COMMODORE & AMIGA nr 12



NR INDEKSU 355216
ISSN 0867-8022

Cena 10 000 zł

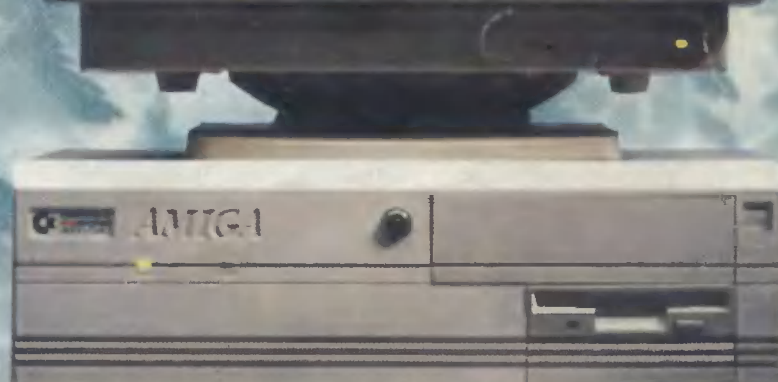
grudzień 1992 r.

MAGAZYN UŻYTKOWNIKÓW KOMPUTERÓW «COMMODORE»





- TEST: AMIGA 600 !
- WYWIAD Z SZEFEM IPS-u
- JAK NAPISAĆ WŁASNE DEMO
- PHOTOKINA RELACJA Z KOLONII





AMIGA 4000

Jak łatwo zauważyć, forma i wygląd naszego pisma zmienia się nieustannie. Staramy się zadowolić jak największe grono Czytelników, stąd ciągłe innowacje. Udało nam się wreszcie przejść na skład komputerowy - na lepsze kolory i papier przyjdzie jeszcze pora. Czas potrzebny na przygotowanie i wydrukowanie numeru uległ nieco skróceniu, co pozwoli nam przekazywać Wam nieco świeższe informacje.

A na komputerowym rynku dzieje się wiele nowego. Firma Commodore lansuje nowy model Amigi oznaczony symbolem 4000, przeznaczony raczej dla profesjonalnych zastosowań. Komputery stają się coraz bardziej skomplikowane, jak będzie wyglądała przyszłość komputeryzacji - czas pokaże. Trzeba przyznać, że oszałamiająca szybkość postępu ma swoje wady. Komputery, które są standardem dziś, już jutro staną się przeżytkiem. Gdzie są te piękne czasy, kiedy to, żeby program działał szybciej, poszukiwało się nowego, lepszego algorytmu, a nie zmieniało komputer na szybszy ...

Wiele złego i wiele dobrego mówi się obecnie na temat Amigi 600, jednego z ostatnich produktów firmy COMMODORE. Nowy KICKSTART 2.04 zainstalowany w A500+ sprawił już dostatecznie dużo kłopotów, jak zatem wygląda sytuacja z ulepszoną wersją systemu oznaczoną symbolem 2.05? Na to pytanie i na wiele innych udzieli odpowiedzi Andrzej Bobek w teście Amigi 600. Ten sam autor zaprasza Was na wycieczkę po targach PHOTOKINA w Kolonii - w obszernym artykule znajdziecie sporo informacji na temat AMIGI 4000.

Zapewne zarówno użytkowników C-64 jak i Amigi ucieszy fakt, że rozpoczynamy dwa cykle dotyczące programów demonstracyjnych. Wydawałoby się, że programy tego typu nie służą niczemu, ale przecież każdy lubi patrzeć, jak programiści wyciskają z biednych scalaczków co tylko możliwe, poza tym to właśnie programy demonstracyjne pozwalają odkryć pełnię możliwości komputera. Autorzy artykułów o demach są od dawna związani z „demosceną", gwarantujemy więc stuprocentową rzetelność informacji.

W imieniu całej redakcji "C&A" pragnę złożyć Wam najserdeczniejsze życzenia
WESOŁYCH ŚWIĄT
i SZCZĘŚLIWEGO NOWEGO ROKU.

Miłej lektury,

BARTŁOMIEJ DRAMCZYK

M • E • N • U

AMIGA:

C-64:

ORAZ:

magazyn użytkowników komputerów « COMMODORE »

Redaktor naczelny: *KLAUDIUSZ DYBOWSKI*
Sekretarz redakcji: *CHRISTIAN GRZENKOWICZ*
Redakcja: *ROBERT CHOJECKI, DARIUSZ DUCKI*
Opracowanie graficzne: *JOLANTA PRZEŹDZIECKA*
Zdjęcia: *JERZY STOKOWSKI*

Stali współpracownicy: *ANDRZEJ BOBEK (szef działu Amigi), RAFAŁ BORZYŃSKI, PIOTR CERKIEWNIK, BARTŁOMIEJ DRAMCZYK, JERZY DUDEK, MARIUSZ FERDYN, PAWEŁ GALAS, BARTŁOMIEJ KACHNIARZ, WOJCIECH KAZIMIERCZAK, ROBERT KULIŚ, PIOTR LISZEWSKI, RAFAŁ PIASEK, OLAF PRZYBYSZEWSKI, BARTOSZ SMAGA*

Redakcja: *ul. Wasilkowskiego 7, 02-776 Warszawa, tel. 643-1840*
Kontakt z Czytelnikami: *piątek w godzinach 13⁰⁰-17⁰⁰*
Wydawca: Spółdzielnia *"BAJTEK", ul. Wspólna 61, 00-687 Warszawa, tel./fax 21-12-05*
Druk: *Prędsiębiorstwo Wydawniczo-Poligraficzne "GRYF", Sp. Akc. Ciechanów*
Nr zlecenia:

Redakcja zastrzega sobie prawo do skracania i adiustacji materiałów. Materiałów nie zamówionych nie zwracamy. Za treść ogłoszeń i/lub reklam redakcja nie odpowiada.



Co dwa lata w Kolonii odbywają się targi Photokina. Poświęcone są one wszystkiemu, co w jakikolwiek sposób związane jest z filmem, telewizją i fotografią. Tegoroczne targi odbywały się w dniach 16-22 września na terenie centrum wystawowego Köln-Messe.

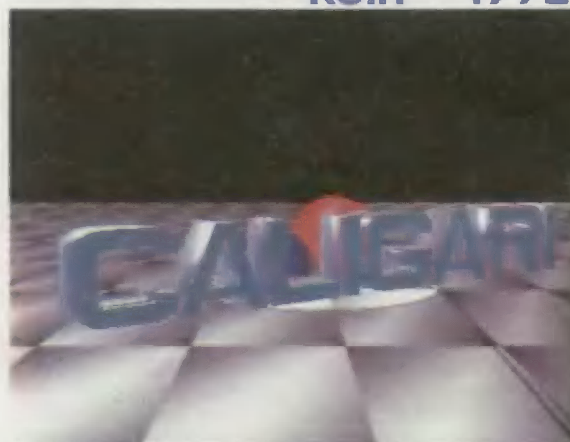
PHOTOKINA SHOW W KOLONII

W czternastu wielkich, dwupoziomowych halach wystawiono wszystko, co miało coś wspólnego (albo i nie) z tematem wystawy. Od latającej, mającej postać helikoptera dwumetrowej kamery, poprzez komputery graficzne, aż po setki urządzeń należących do tzw. elektroniki domowej. Zwiedzenie całej wystawy zajęło mi trzy dni, przy czym pod koniec trzeciego natrafiałem jeszcze na miejsca, w których byłem pierwszy raz.

Na targach obecne były, oprócz setek niewielkich, specjalizowanych firm, także wszystkie wielkie koncerny, jak Panasonic czy Sony. Ten ostatni zaprezentował swój najnowszy wynalazek, tzw. Mini-Disk, reklamowany jako „istna rewolucja". Chyba nie bez racji, jest to bowiem około 2,5-calowa optyczna dyskietka, na której możemy zapisywać (tak, zapisywać) mniej więcej 70 minut muzyki z jakością CD. Sony przedstawiło już oczywiście wszelkie urządzenia wykorzystujące Mini-Disk, jak choćby Walkman, radio samochodowe czy przenośny radiomagnetofon. Konkurencja promowała natomiast magnetofony DCC (Digital Compact Cassette), stosujące zapis cyfrowy. Doprawdy, zapowiada się ciekawa rywalizacja.

Wszystkie liczące się firmy mające coś wspólnego z telewizją prezentowały telewizory HDTV, czyli telewizji wysokiej rozdzielczości. Było też nieco firm promujących swoje wynalazki w dziedzinie telewizji trójwymiarowej.

Wśród setek prezentowanych wytworów myśli ludzkiej niemało było naprawdę robiących wrażenie. Jedna z firm obecnych na wystawie zaprezentowała na przykład gigantyczny, półkolisty telewizor, obejmujący swym łukiem wycinek okręgu o promieniu 120 stopni. Wyświetlano na nim między innymi materiał nakręcony z helikoptera latającego kilkanaście metrów nad szczytami Alp. Trudno się było oderwać.







PRZYKŁADOWE MOŻLIWOŚCI PROGRAMU CALIGARI

Spore wrażenie zrobiła też na mnie hala poświęcona samochodowemu sprzętowi grającemu. Nie wyobrażałem sobie dotąd, jak można w średniej wielkości samochodzie zmieścić sprzęt grający z kilku dyskotek. Teraz już wiem, że można. Jedna z firm postanowiła bowiem swoim sprzętem, zainstalowanym w Volkswagenie Golfie, nagłośnić całą halę - i udało się jej.

Cała impreza była zorganizowana z olbrzymim rozmachem - najlepszym przykładem może tu być „stoisko" koncernu Sony. Firma ta wykupiła bowiem spory kawał hali i... zamieniła go w ulicę Nowego Yorku. Nie pominięto niczego: był McDonald, Michael Jackson, Cadillac, stojący na chodniku i grający na saksofonie facet. Co jakiś czas z ukrytych głośników dobiegały syreny wozów policyjnych. Nie zapomniano też o nieco innych warunkach panujących w Nowym Yorku - „uliczka", znajdująca się w środku sterylnie czystej (w stylu niemieckim) hali, była dokładnie usłana petami i brudnymi papierami.

Cóż, to by było na tyle o wystawie z szerszej perspektywy, pora na komputerowe konkrety.

AMIGA

Amiga jest na całym świecie znana jako komputer wspaniale nadający się do dwóch rzeczy: do gier i do grafiki. Co do gier, to nawet i ta sfera zastosowań była widoczna na wystawie - koncern Sony wyprodukował grę komputerową na swą cześć, oczywiście w wersji dla Amigi.

I to było jedyne nieprofesjonalne zastosowanie Amigi. W pozostałych, bardzo licznych miejscach, była ona jak najpoważniej wykorzystywana do najróżniejszych zadań. Najwięcej widziało się Amig służących do prezentacji komputerowej, czyli do generowania na wielu ekranach na raz napisów i grafiki opisujących zalety firmy, na której stoisku Amiga stała. Do ciekawszych zaliczyłem stoisko firmy Mitsubishi, gdzie na całej ścianie monitorów wyświetlana była reklama wychwalająca genialne „pecety" Mitsubishi - z tyłu zaś stała sobie spokojnie Amiga i całą rzecz realizowała.

Tak więc Amigi były na wystawie UŻYWANE. Były jednak także PREZENTOWANE i to w wielu miejscach. Amig używało bowiem sporo firm trudniących się reklamą. Najgorsze wrażenie wywarło na mnie stoisko pewnej firmy, gdzie z magnetowidu puszczano animacje, stworzone co prawda z użyciem Imagine'a, jednak tak nieumiejętnie, że gotów byłem uwierzyć, że facetów opłaca konkurencja.

Dalej było już dużo lepiej. Spore stoisko wykupiła firma AEON, sprzedająca w Europie Zachodniej program animacyjny Caligari Broadcast 2. Na wielkim ekranie pokazywano gotowe animacje, zaś z boku stała maksymalnie rozbudowana Amiga 2000, przy której siedział sobie pan, gotów każdemu zaprezentować program. Można było się naprawdę wszystkiego dowiedzieć, wszystko zabaczyć i o wszystko zapytać. Pokazano mi między innymi wersję beta Caligari Broadcast 3 i muszę przyznać, że zrobiła ona na mnie wrażenie. Wyposażony w nowe funkcje edytor obiektów jest bardzo dobry, z pewnością lepszy od tego, który oferuje nam Imagine, wprowadzono też możliwość definiowania kilku bardzo przydatnych parametrów pracy modułu liczącego ostateczną wersję grafiki.

Duże stoisko, zawalone kilkunastoma Amigami 3000, zajmowała firma Commodore. Nie była tam zresztą sama. Towarzyszyła jej na przykład firma Videocomp, promująca najnowszą wersję programu Scala - Scalę 2.00, nazwaną MultiMedia 2.00, w skrócie MM200. Nowa wersja jest rzeczywiście rewelacyjna, bije konkurencję na głowę - niewiele istnieje dla Amigi tak rozbudowanych programów. A jeśli ktoś nie wie, do czego tak właściwie jest program Scala 2.00, znany w Polsce z wcześniejszych wersji, służę wyjaśnieniem. Umożliwia on generowanie napisów, nakładanie ich na wszelakiego rodzaju grafikę, dodawanie do tego animacji, muzyki lub też samych sampli. Wszystko to może być mieszane na setki sposobów, na przykład napis na ekranie może pojawiać na jeden z czterdziestu sposobów. Program oferuje też oczywiście możliwość sterowania urządzeniami zewnętrznymi, na przykład odtwarzaczami laserowych płyt wizyjnych. Wszystko to określa się jednym słowem: MultiMedia. I czego by nie wygadywano na innych stoiskach, Amiga z uruchominą Scalą robiła najlepsze wrażenie. Poza tym firma Videocomp naprawdę umiejętnie wzięła się za reklamę rozpowszechnianego przez siebie produktu. Nie skąpiono reklamówek, każdy, kto wykazał jakiekolwiek zainteresowanie, był wszechstronnie informowany i obsypywany nowymi prospektami. Jeden z nich stanowiła lista kilkuset poważnych użytkowników programu na całym świecie, rzeczywiście robiąca wrażenie. Nikt nie był ignorowany, bez kłopotów porozmawiałem sobie na przykład z jednym ze współtwórców programu, panem Einarem Haugstadem. Nie miał wesołej miny, gdy opowiedziałem mu o polskim „rynku" oprogramowania.

Na tymże samym stoisku, firmowanym przez Commodore, można było pobawić się najnowszą kartą 24-bitową dla Amigi - Opal Vision. Do karty dołączany jest doskonały program, Opal Paint. Do końca roku gotowe też będą dodatkowe moduły do karty: genlock, frame-grabber, kość odpowiedzialna za efekty specjalne, jak na przykład owijanie obrazu w czasie rzeczywistym na kuli.

Na stoisku stał też dziwny, przez większość ludzi ignorowany komputer. Nie przypominał on żadnej Amigi, jego przód wyglądał zresztą na ręcznie robiony. To właśnie była Amiga 4000. Dopiero dwa dni przed końcem wystawy sprowadzono skądś tę nowość w wyjątkowo brzydkiej obudowie. W Pasadenie, gdzie miała miejsce jej amerykańska premiera, Amiga 4000 wzbudziła sensację - tutaj stała sobie cichutko i wyświetlała obrazki, demonstrując swe nowe możliwości graficzne. Nie ma się jednak co dziwić - przecież nie była to wystawa poświęcona Amidze, czas A4000 nadejdzie podczas targów we Frankfurcie.

Amiga 4000 jest konstrukcją, moim zdaniem, rewelacyjną. Dzięki niej Amiga, tak jak siedem lat temu, znowu zdystansuje konkurencję swoimi możliwościami graficznymi, choć wątpię, by na dłużej niż dwa lata - do tego czasu 24 bity powinny stać się standardem.

Ale po kolei. Prezentowana A4000 miała jeszcze starą kość muzyczną - nowa, 16-bitowa, nie jest jeszcze gotowa. I ta kość to było właściwie wszystko, co zostało z poprzednich Amig. A4000 wyposażona jest w procesor Motorola 68040 taktowany zegarem 25 MHz, jest więc mniej więcej trzy razy szybsza od Amigi 3000. Ma się w niej instalować minimum 4 MB RAM (2 megabajty instalowane w A3000 to śmiesznie mało), 120-megabajtowy twardy dysk, prawdziwą, szybką „gęstą" stację dysków.

Nowa Amiga, z tego, co na razie wiadomo, nie dysponuje nowymi rozdzielczościami - stare uznano podobno za wystarczające. Tak więc minimum to 320*256, maksimum - 1280*512. Stare wysłużone kostki Agnus i Denise zastąpiono nowymi, o imionach Lisa i Alice.



Amiga 4000

Paletę rozszerzono z 4096 do 16.777.216 kolorów, zmieniono też zasadę fuckcjonowania kości specjalizowanej. Liczba wyświetlanych kolorów nie zależy już od rozdzielczości, jak to ma miejsce w „starych" Amigach. Wracając do kolorów: A4000 potrafi wyświetlać ich na ekranie 2, 4, 8, 16, 32, 64, 128, 256, 4096 lub 262.144 - wszystkie z palety 16,8 mln. Ostatni z wymienionych trybów działa na podobnej zasadzie jak swego czasu oszałamiający HAM - nazwano go też podobnie, a mianowicie HAM8, jako że opiera się na ośmiu bitach. Trzeba przyznać, że wyświetlane na Amidze 4000 grafiki były nie do odróżnienia od tych generowanych przez OpalVision.

Na razie jest to niestety wszystko, co wiadomo o A4000, na więcej trzeba nam widać poczekać do wystawy we Frankfurcie. W każdym razie zapewniono mnie, że Amiga 4000 będzie gotowa do masowej produkcji do końca roku.

Na stoisku Commodore stały jeszcze inne komputery - a mianowicie Silicon Graphics. Co ciekawe, Amigi 3000 były posprzęgane z Siliconami i dość sprawnie razem sobie radziły. Wszędzie też walały się reklamówki programu animacyjnego dla Siliconów o nazwie 3D-GO, który - wedle wypowiedzianych szeptem słów jednego z organizatorów - na początku przyszłego roku ma zostać wypuszczony w wersji dla Amigi. Zbyt piękne, by mogło być prawdziwe.

NIE TYLKO O AMIDZE

Muszę przyznać, że ze sporą satysfakcją odnotowałem zerowy niemal udział „pecetów" wśród komputerów używanych do grafiki i jakichkolwiek zastosowań z dziedziny multimedia w ogóle. Jedynym chyba ich zastosowaniem, jakie widziałem na wystawie, było sterowanie wszelakiej maści magnetowidami i innymi urządzeniami. Słowo „satysfakcja" nie pojawiło się tu jednak dlatego, że jestem jakimś maniakalnym wrogiem firmy IBM. Po prostu do szału doprowadzają mnie reklamy w stylu „PC - true multimedia". Obawiam się tylko, by nie stało się po myśli ludzi, którzy takie reklamy układają - nie pierwszy już raz wygrałoby nie lepsze, ale lepiej sprzedawane.

Ale komputery to oczywiście nie tylko Amiga i klony. Na całej, gigantycznej wystawie widziałem JEDNO Atari ST. Sporo było natomiast Macintosh'y. Około pięćdziesiąt procent robiło dobre wrażenie wyświetlając kolorowe, 24-bitowe obrazki, reszta zaś była używana do najróżniejszych celów. Oprócz DTP, do którego zatru-

dniali Macintosh'e ludzie z firm zajmujących się fotografią, zastosowania tych komputerów były nieco podobne jak w przypadku Amigi - czyli grafika. O ile jednak najczęściej Amigę widywało się w roli komputera do tworzenia wystawowych reklamówek, o tyle to zastosowanie w przypadku Macintosh'a zdawało się być marginalnym, podobnie jak DTP w przypadku Amigi.

Mniej więcej tyle samo było na wystawie firm reklamujących Macintosh'e jako komputery do animacji trójwymiarowej, jak i w przypadku Amigi. Ze smutkiem muszę jednak stwierdzić, że - jak zwykle - firma Apple lepiej radzi sobie z reklamą niż Commodore. Chwilami co prawda, jak na mój gust fachowca, przesadzano, jak na przykład miało to miejsce w przypadku „pokazu rewelacyjnych możliwości komputerów Macintosh". Za pomocą rzutnika obraz z komputera wyświetlany był na wielkiej planszy - i rozmach był jedyną rzeczą, która mogła mi zaimponować. Otóż te „rewelacyjne możliwości" oscylowały gdzieś w okolicy Deluxe Paint'a IV, tyle że w grafice 24-bitowej. Cały czas nasuwało mi się jednak pytanie: dlaczego firma Commodore nie może robić takiej krzykliwej reklamy Amidze? Przecież Apple wypromował swoje komputery w co najmniej 50 procentach dzięki dobrej reklamie.

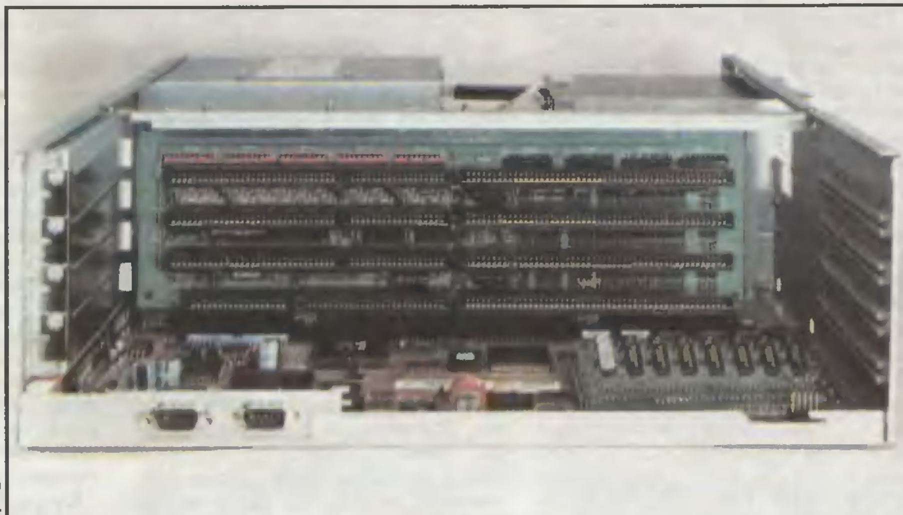
Na wystawie można też było obserwować konkurowanie ze sobą Amigi CDTV i bardzo podobnego wynalazku firmy Philips, nazwanego CD-I (Compact Disc Interactive). O ile promowanie CDTV przez Commodore kończyło się właściwie na wystawieniu kilku egzemplarzy na widok publiczny i rozdawaniu płyt przeznaczoncych dla CDTV, to Philips reklamował swój produkt z większym rozmachem - na przykład umieścił na stoisku z CD-I kilkumetrową kopię pilota do tej maszyny, ustawił też kilka egzemplarzy CD-I tak, by co młodsi zwiedzający mogli sobie pograć.

PODSUMOWANIE

Photokina była niezwykle ciekawą wystawą, zarówno dla mnie, fana komputerów z odchyleniem w kierunku grafiki, jak i dla tysięcy zwykłych ludzi codziennie ją zwiedzających. Każdy mógł coś dla siebie znaleźć: ja - Amigę, dzieciaki - gry telewizyjne, fachowcy z filmu - kompletne wyposażenie planu filmowego itd itd.

Zaś co do Amigi, to - mówiąc szczerze - spodziewałem się, że nie będzie jej prawie widać, że utonie w gąszczu dalekowschodniej tandety. Widać jednak, że jakoś się sama broni, tak jak to się dzieje od siedmiu lat, podczas których firma Commodore w najlepszym wypadku nie robiła nic, by Amigę wykończyć. Miejmy nadzieję, że ostatnie, sprawiające wrażenie rozsądnych, poczynania tej firmy zapowiadają trwałą zmianę jej polityki.

ANDRZEJ BOBEK



Amiga 4000 od wewnątrz

NIEOFICJALNE DANE TECHNICZNE AMIGI 4000

CPU (główny procesor):
- Motorola 68040, 25 MHz

PAMIĘĆ:
- 2 MB CHIP-RAM
- 2 MB FAST, rozszerzalna na płycie (moduły SIMM) do 16 MB

SYSTEM OPERACYJNY:
- 512 KB, ROM, Kickstart 3.0

ZŁĄCZA:
- klawiatury
- 2 * Mysz/Joystick
- szeregowe (RS-232)
- równoległe (Centronics)
- monitora (RGB lub RGBI)
- dwóch kanałów stereo
- stacji dysków
- wewnętrzny port dla napędów AT-BUS
- port monitora VGA (31 kHz)

WEWNĘTRZNE ZŁĄCZA SYSTEMU:
- CPU (procesora) - 200 linii
- cztery złącza kart rozszerzeń w standardzie Zorro III (100 linii) - trzy w standardzie AT
- specjalizowane złącze wizyjne

Uwaga: istnieją cztery linie złącz (po dwa typy w linii - trzy linie „Zorro III + AT" i jedna „Zorro III + video"), w praktyce oznacza to, że na raz można podłączyć tylko cztery karty.

KLAWIATURA:
- odłączalna
- 94 klawisze, wydzielony blok numeryczny
- w zasadzie identyczna jak w A2000 lub A3000

MYSZ:
- optyczno-mechaniczna, dwa przyciski

STACJE DYSKÓW:
- wbudowana „gęsta" stacja 3,5 cala
- pojemność od 880 KB do 1,76 MB
- możliwość wewnętrznego montażu dwóch dodatkowych stacji dysków: 3,5 i 5,25 cala

TWARDY DYSK:
- od 120 MB w górę
- typ: AT-BUS, w przyszłości przewidywany też SCSI

GRAFIKA:
- nowy zestaw kości specjalizowanych AGA (nazywany też AA)
- rozdzielczości od 320*256 do 1280*512 (nie licząc obszaru ramki)
- przełączalność NTSC/PAL
- 24-bitowa paleta kolorów, 16.777.216 możliwości
- 2 do 262.144 kolorów na raz na ekranie

TRYBY PRACY WYJŚĆ DLA MONITORA:
- zwykły monitor RGB lub VGA (niektóre tryby graficzne dostępne tylko z monitorem VGA)
- częstotliwość pozioma: 15-31 kHz
- częstotliwość ramki: 50-72 Hz

DŹWIĘK:
- cztery 8-bitowe przetworniki, po dwa na kanał stereo

WYMIARY:
- 15,25 cala głęboka, 15 cali szeroka i 5 cali wysoka



DEMOSY (1) - CO TO JEST I PO CO TO KOMU?

Dzięki drobnej reformie w piśmie C&A możecie wreszcie poczytać o tym, do czego naprawdę jest stworzona Amiga. Tak, mogę z czystym sumieniem powiedzieć, że Amiga jest najlepszym na świecie sprzętem pod względem ilości i jakości programów zwanych potocznie DEMKAMI, DEMOSAMI czy też DEMÓWKAMI. Ale zacznijmy od początku, czyli co to jest demo?

Demo, czyli inaczej program demonstracyjny, jest czymś w rodzaju filmu stworzonego na komputerze. Powiedzmy raczej teledysku, gdyż na ogół mamy do czynienia z różnego rodzaju efektami graficznymi dopasowanymi do wspaniałej (jak zwykle na Amidze) muzyki. No tak, powiecie, ale przecież o wiele lepsze filmy można obejrzeć na VIDEO, a teledyski w MTV. Jeśli tak właśnie myślicie, to w zasadzie nie ma sensu, abyście czytali ten cykl artykułów. Sądzę również, że niezbyt interesujecie się komputerami, a jeżeli nawet, to jest to etap chwilowy i z pewnością szybko wam przejdzie. Dlaczego tak myślę?

Gry szybko się znudzą, a siedzenie przy jakimś „użydzie" (programie użytkowym) trudno nazwać zabawą. Nie chcę tu wcale udowadniać, że oglądanie demówek nigdy się nie znudzi, ale jeżeli jesteście AMIGERAMI z prawdziwego zdarzenia (a sądzę, że tak, skoro czytacie C&A), to na pewno rozumiecie, o co mi chodzi - dema to po prostu klasa (czy jak kto woli - sztuka, jakość) sama w sobie. U ludzi zaangażowanych osobiście w tworzenie dem (tzn. związanych ze sceną) każde nowe demko wyzwala twórcze zapędy - natychmiast zastanawiają się, jakby tu zrobić samemu coś podobnego. Kto wie, może dzięki demkom odkryjecie w sobie talent i zostaniecie wspaniałymi muzykami, grafikami lub tym najważniejszym, czyli koderami (programista piszący programy w asemblerze). Naprawdę możecie mi wierzyć, że tworzenie demosów to wspaniała zabawa, a przy okazji można zaprzyjaźnić się z wieloma ludźmi na całym świecie.

A więc wiecie już, co to są demosy i dlaczego są godne uwagi, teraz wypadałoby wytłumaczyć, kto robi takie cudka i po co? Odpowiedzieć na pierwsze pytanie nie można jednym zdaniem, gdyż sprawa jest trochę bardziej skomplikowana. Pierwsze demka na AMIGĘ wypuściła firma COMMODORE (a jakże by mogło być inaczej). Te prymitywne (w porównaniu do obecnie najlepszych) demosy były firmie potrzebne do prezentacji możliwości danego typu komputera na wszelkich targach komputerowych, wystawach itp. Oczywiście w tamtych czasach AMIGA przyciągała wszystkich ze względu na rewelacyjne możliwości graficzne i muzyczne. Z tego też powodu już pod koniec roku 1987 zaczęła powstawać amigowska „scena".

W tym miejscu doszliśmy do punktu kulminacyjnego tego artykułu, bo kto zrozumie, co to jest scena, będzie już na pewno wiedział, kto i po co robi dema (chociaż często sami autorzy nie wiedzą, po co je robią). Sceną nazywamy wszystkie grupy komputerowe zrzeszające koderów, muzyków, grafików, swapper'ów, a obecnie i modem-trader'ów oraz sysop'ów, a ich celem jest crack'owanie oryginalnych programów, pisanie programów użytkowych, demosów (a czasami, dla kawału, również wirusów) i rozpowszechnianie tego na całym świecie. Chyba trochę za dużo nowego słownictwa jak na jedno zdanie. Teraz po kolei spróbuję wytłumaczyć słowa, którymi wam przed chwilą tak zamąciłem. Oczywiście zacznę od końca, żeby było najprościej (he,he).

CRACKOWANIE (łamanie) ORYGINAŁÓW

Wiadomo, że oryginalne programy na Amigę (i nie tylko) nie należą do najtańszych, a że Amiganci są zachłanni i lubią mieć dużo małym kosztem, więc kopiują dyskietki bez zważania na przepisy prawne (skąd my to znamy? X-COPY?). Tak, ale w przypadku oryginalnych programów (weźmy na przykład gierki) nie jest to takie proste i nie wystarczy zwykle X-COPY. Wiadomo, że firmy software'owe niezbyt lubią, kiedy nielegalnie rozpowszechnia się ich programy i dlatego rozpoczęły walkę z piratami komputerowymi, a dokładniej z grupami crackerskimi poprzez zabezpieczanie oryginałów w rozmaity sposób. Ale wiadomo, że co da się zabezpieczyć, można i odbezpieczyć. Koderzy chcąc się pochwalić, że to właśnie im udało się złamać wspaniałą gierkę, zaczęli dorabiać do nich własne czołówki, będące niczym innym jak pierwszymi demkami zwanymi INTRAMI (INTRO to skrót od INTRODUCTION czyli początek, wstęp). Sam koder jednak na ogół sam nie był w stanie zrobić wszystkiego, dlatego musiał dogadywać się z grafikami czy muzykami. I w ten oto sposób zaczęły powstawać pierwsze „demo-grupy". Zaczęła się tworzyć rywalizacja między nimi, skutkiem czego wytworzyła się ELITA.

CODER - programista piszący programy w asemblerze. Dawniej najczęściej używano SEKI (nazwa asemblera), a obecnie najpopularniejszym programem tego typu na scenie jest chyba TRASH'M'ONE ASSEMBLER. Zadanie kodera to pisanie demek (nie tylko wektorówki, jak niektórym się wydaje), czasami również programów użytkowych i to BARDZO DOBRYCH, czego przykładem może być PROTRACKER (Amiga Freelancers), SEKA (Kefrens), MASTER SEKA (Spread Point), ASM'ONE (Kefrens), TRASH'M'ONE (Silents), IFF CONVERTER (Kefrens), DISK MASTER 3 (Paradox), NOISE PACKER (Phenomena) i cała masa innych (w nawiasach podane są nazwy grup, z których wywodzą się autorzy tych programów).

CRACKER - inaczej pirat, czyli ten od łamania oryginałów. Jest to odmiana kodera potępiana przez wszystkich użytkowników oryginalnego oprogramowania. W obecnej chwili cracker na ogół nie robi INTEREK, gdyż grupy są na tyle dobrze zorganizowane, iż może to za niego zrobić koder. Cracker jednak robi czasem TRAINERY czyli menu z możliwymi nieśmiertelnościami do gry (a czyż nie lubimy wygrywać w grę od razu za pierwszym razem?).

Istnieją jeszcze koder-crackerzy, zajmujący się poprawianiem BUG'ów (błędów) w cudzych programach. Najczęściej poprawiają wersje licencjonowanych użytków (raczej tych napisanych w asemblerze), przerabiają programy tak, aby działały na innych typach AMIGI, wykorzystywały lepiej dodatkową pamięć itp. Ludzie ci robią to zupełnie bezinteresownie, nie czerpiąc z tego żadnych profitów, mimo to są najbardziej gnębieni przez „łowców piratów".

SUPPLIER. Zastanawialiście się, skąd cracker bierze oryginały do łamania? Tym właśnie zajmuje się supplier - błyskawiczne dostarcza najnowsze programy crackerowi. Proces ten w dobie modemów i tak ogromnej konkurencji musi być naprawdę szybki, gdyż różnica jednego dnia może spowodować wydanie tego samego programu przez kilka innych grup. Supplierzy są często pracownikami firm software'owych, dzięki czemu mogą szybciej od innych dostać najnowszy produkt (nie raz już się zdarzyło, że wersja zcrackowana została rozpowszechniona szybciej niż oryginał!!).

GRAFIK (GFX-ARTIST). Chyba rozumiecie, czym zajmuje się taki człowiek? Nie rysuje on jednak obrazów, chociaż obecnie w demkach wszystko może się zdarzyć. Głównym zajęciem grafika w grupie robiącej demka jest robienie:

- FONTÓW (czyli zestawów znaków, zazwyczaj dużych i kolorowych, choć nie jest to regułą),
- LOGOSÓW (czyli niesamowitych napisów z nazwą grupy, demka itp.; mogą one być małe (np. mały poziomy pasek ekranu), jak i wielkie - loga na cały ekran z obrazkiem w tle),
- OBRAZKÓW CAŁOEKRANOWYCH (fullscreen pictures), IKON, TŁA, ANIMACJI (rzadziej spotykane i raczej tylko w demosach, choć właśnie te dzieła przynoszą grafikom największą sławę, oczywiście jeżeli są w tym naprawdę dobrzy).

Wśród grafików ze sceny przyjęty jest zwyczaj NIE SKANOWANIA swoich obrazków, jak również nie używania digitalizacji! Klienci, którzy dopuszczą się takiego występku (podpisując się pod nim jak pod własnym dziełem) skazani są na bluzgi ze strony całej sceny i wieczne potępienie. Niektóre grupy wydają demka z digitalizowanymi zdjęciami oczywiście nie podpisując się pod nimi. Demka te noszą nazwę SLIDESHOW-ów i nie biorą udziału w konkursach razem z „normalnymi demkami". Dobrzy (i jednocześnie pracowici) graficy wydają czasami demka w stylu slideshow'a, jednak wszystkie zawarte w nich obrazki są ich oryginalnymi dziełami stworzonymi na ukochanej AMISI.

Najpopularniejszym programem (standardem) graficznym używanym przez grafików ze sceny jest DELUXE PAINT (aktualnie (8'92) najnowsza wersja 4.1).

MUZYK (musician). Z pewnością najbardziej zrozumiała przez was funkcja. Zadaniem muzyka jest stworzenie ścieżki muzycznej do demka (intra, music-disk'u, slideshow'u itp.). Skomponowanie ładnego kawałka przy pomocy komputera z pewnością wymaga sporego talentu (i przede wszystkim słuchu). Mimo to muzycy stanowią dość liczny naród i należą do najproduktywniejszych ludzi w grupie (w końcu do dema muzyk na ogół robi jeden lub dwa utwory, a grafik musi zrobić całą listę grafiki). Czasami muzycy robią również efekty dźwiękowe do animacji i innych takich, jednak zdarza się to rzadko i dlatego o tym pisać nie będziemy.

Warto natomiast na dłużej zatrzymać się przy demkach muzycznych - music-disk'ach (dawniej sound-disk'ach). Są to dema, których główną atrakcją jest właśnie muzyka. Zwykle mamy do wyboru kilka modułów (utworów), które możemy sobie posłuchać wybierając dowolny z nich. Liczba modułów w takim demku waha się od kilku (np. TURMOIL) do ponad stu (HIS MASTERS NOISE)!! Ponieważ muzycy, jak już wcześniej wspomniałem, stanowią rasę najpracowitszą, tak więc music-disk'i należą do demosów często spotykanych w naszych (maniaków) zbiorach. Co prawda nie biorą one udziału w competitions (konkursach), za wyjątkiem dwóch ostatnich music-demo-disk'ów SANITY, ale za to można je zobaczyć w magazynach dyskowych w chartsach (specjalnej liście przebojów).

Podobnie jak DPaint dla grafików, dla muzyków standardem jest program PROTRACKER (najnowsza wersja 2.2BETA). Muzyczki w nim zapisywane mają postać modułów i dlatego utwory te nazywa się potocznie modułami, a muzyków - modułowcami.

SWAPPER. No dobrze, crackerzy crackują, koderzy kodują, ale przecież nie robią tego tylko dla siebie (po co w końcu robi się demka). Ktoś przecież musi to wszystko rozpowszechnić, porozsyłać do przyjaciół (kontaktów) z innych grup. Tym kimś jest właśnie mail-swapper trudniący się... swapowaniem.

Co to jest swaping? Jest to korespondencja listowna z innymi grupami na całym świecie służąca wymianie najnowszych programów, demek. Człowieka, z którym koresponduje swapper, nazywamy kontaktem (contact - to od kontaktowania się). Dobry swapper posiada kilkadziesiąt kontaktów. Po scenie krąży opinia, że czym lepszy swapper, tym krótsze pisze listy. Jednak z najnowszych sondaży w zachodnim magazynie dyskowym R.A.W. wynika, że nie jest to prawdą, gdyż obecnie swapuje się głównie po to, by zdobyć nowych przyjaciół. Jest to następstwo tego, że funkcja swappera jest już „na wymarciu". Dlaczego? Dlatego, że o wiele lepszym, szybszym i pewniejszym sposobem na przesyłanie programów jest MODEM (urządzenie przesyłające programy poprzez sieć telefoniczną). Najśmieszniejszą rzeczą w tym wszystkim jest fakt, że w chwili, kiedy funkcja swappera wypierana jest przez modem-trader'ów, to w Polsce swapperzy zaczynają się mnożyć i krąży powszechna opinia, że dzięki nim dogonimy scenę zachodnią (w sumie jest w tym tyle prawdy, że przynajmniej swapperzy muszą nauczyć się angielskiego, a to na pewno we wszystkim pomoże).

MODEM-TRADER - obecnie najpopularniejsza funkcja na świecie (w Polsce dopiero rozwijająca się, i to jedynie w stolicy). Na zachodzie praktycznie prawie każdy amigowiec posiada w domu modem. Dzięki temu wynalazkowi obecnie wszystkie najnowsze programy rozchodzą się po świecie w ciągu jednego dnia! Modem-trader to człowiek, który żyje z wysyłania (UPLOAD'owania) najnowszych produkcji swojej grupy (i nie tylko) do BBS'ów i w zamian ściągania (DOWNLOAD'owania) najnowszego stuff'u (programów) dla siebie i sąsiadów. Co to są jednak BBS'y? Dowiesz się jak przeczytasz, kto to jest SYSOP. Najczęściej używanym programem przez modem-trader'ów i sysop'ów jest AMI-EXPRESS (+ programy archiwizyjące).

SYSOP - również funkcja związana z modemami. Sysop jest jednak człowiekiem o wiele ważniejszym i potrzebniejszym niż mo-

ciąg dalszy na str. 36



AUTOMATYCZNY PRZEŁĄCZNIK MYSZ/JOYSTICK

Rodzina *gadgetów* do Amigi ciągle się powiększa, czego dowodem jest pojawienie się na rynku przełączników mysz/joystick oferowanych przez różnych dystrybutorów. Ja dostałem do testowania dwa takie przełączniki: pierwszy, produkcji niemieckiej, od firmy 3-STATE POLAND i drugi, krajowy, z firmy Mapasoft.

PRZEZNACZENIE

Przełącznik umożliwia podłączenie do jednego portu jednocześnie dwóch urządzeń: myszy i joysticka. Dzięki temu unikamy trochę niebezpiecznego podłączania joysticka podczas pracy komputera, gdy zapragniemy np. zagrać z przyjacielem w grę na dwa „dżoje". Urządzenie, którym chcemy sterować, wybiera się naciskając FIRE w joysticku lub lewy klawisz myszy i można to robić nawet podczas gry.

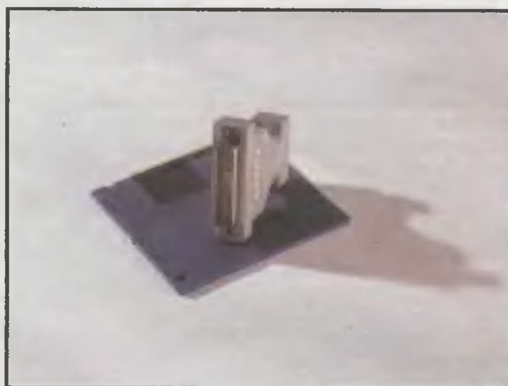
MONTAŻ

przełącznika jest banalny. Jedną stroną wkładamy go do portu myszy (lub joysticka - jeżeli to potrzebne), zaś z drugiej strony podłączamy mysz i joystick. Ważne jest, do którego złącza przełącznika podłączymy oba urządzenia. Mysz powinna być podłączona z prawej strony, a joystick z lewej. W przeciwnym przypadku joystick nie będzie pracował poprawnie (na myszy zmiana miejsca podłączenia nie robi wrażenia).

RÓŻNICE

Jak już wspomniałem, do testowania otrzymałem dwa przełączniki, mogłem więc pokusić się o ich porównanie. Różnice ograniczają się do spraw mało lub bardzo mało istotnych, jak na przykład kolor (kształt obu jest identyczny, zupełnie jakby obudowa pochodziła z tej samej formy), niemniej kilka rzeczy warto odnotować.

Na przełączniku produkcji polskiej (firma Mapasoft) znajduje się naklejka informująca, które urządzenie gdzie podłączyć. Natomiast do przełącznika pochodzącego z firmy 3-STATE POLAND (produkcja niemiecka) dołączona jest kartka



Przełącznik mysz/joystick firmy Mapasoft z Krakowa

sugerująca, jakoby kolejność podłączenia była bez znaczenia, co okazało się nie do końca prawdą. Ponadto krajowy przełącznik ma dwie śrubki pozwalające przykręcić go do komputera. Początkowo myślałem, że z powodu konstrukcji urządzenia jest to niewykonalne (nikt nie produkuje przegubowych śrubokrętów), ale przeprowadzenie próby udowodniło, że można sobie z tym poradzić za pomocą normalnego śrubokręta.

Pod względem działania rożnica jest tylko jedna. Po starcie systemu przełącznik niemiecki zachowuje się tak, jakby go nie było - wskaźnik myszy reaguje natychmiast. Natomiast w przypadku przełącznika polskiego wskaźnik myszy stoi w miejscu. Dopiero po uaktywnieniu przełącznika naciśnięciem lewego klawisza myszy reaguje on na ruchy „gryzonia".

Z przełącznikiem firmy Mapasoft miałem z początku problemy, a mianowicie nie mogłem go podłączyć do komputera. Okazało się, że styki wewnątrz gniazda (tego, które się wkłada do komputera) były za bardzo ściśnięte. Przez to bolce portu myszy trafiały na krawędź styku zamiast do jego środka. Drobna korekta przeprowadzona za pomocą cyrkla pozwoliła na przystąpienie do dalszej części testów. Mam nadzieję, że ta drobna wada jest cechą tylko tego egzemplarza (na styku „made in Unitra" widnieje przecież znak jakości Q).

PODSUMOWANIE

Przełącznik jest dedykowany użytkownikom, którzy z powodu szczupłości miejsca mają komputer „wprasowany" w regał czy biurko - w takim przypadku banalne podłączenie joysticka urasta do rangi problemu. Nie oznacza to, że omawiane tu przełączniki nie przydadzą się innym użytkownikom, a zwłaszcza tym nawiedzanym przez miłośników komputerowej piłki nożnej, koszykówki lub tenisa.

PAWEŁ GALAS

Przełącznik produkcji polskiej:

DYSTRYBUTOR:
PROABIT, 05-500 Raszyn k/Warszawy ul. Mickiewicza 14, tel. 56-08-91

ZALETY
+ możliwość przykręcenia do komputera
+ jasny opis podłączenia (na przełączniku)

WADY
- konieczność uaktywnienia myszy po starcie systemu

Przełącznik produkcji niemieckiej:

DYSTRYBUTOR: 3-STATE POLAND, 02-695 Warszawa, ul. Orzycka 20 tel. 43-16-56

ZALETY
+ automatyczne uaktywnienie myszy po starcie systemu

WADY
- myląca instrukcja
- brak informacji na urządzeniu

ROZSZERZENIE PAMIĘCI A504 PLUS

CZY JESTEM SZALEŃCEM?

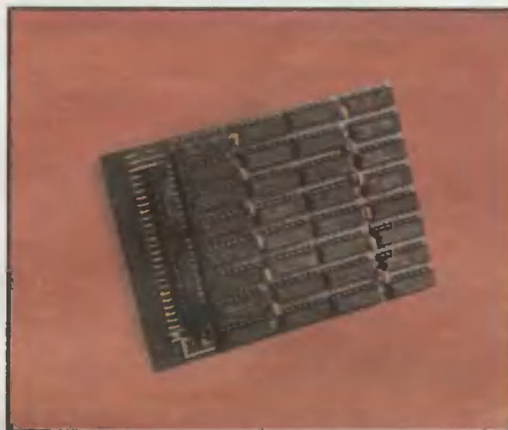
Wczoraj otrzymałem do testowania rozszerzenie pamięci A504 PLUS. Muszę powiedzieć, że przy pierwszym kontakcie nie zachwyca ono chyba nikogo. W tekturowym szarym pudełku, na którym ktoś postawił pieczątkę z nazwą rozszerzenia, znajduje się niewielka płytka. Rzuca się jednak w oczy olbrzymia ilość układów scalonych. Wyobraźcie sobie, że jest ich aż 34, z czego 32 to kości RAM! Przed zainstalowaniem w komputerze tego rozszerzenia spod klapki wyciągnąłem inne. Było to rozszerzenie firmy Alfa Data, na którym jest tylko osiem kości. "To ciekawe..."mruknąłem pod nosem i nadziałem nowe „cacuszko" na igiełki portu Amigi. Tradycyjnie zasłaniając oczy przed ewentualną eksplozją (człowiek testujący hardware to wielki ryzykant - naraża komputer oraz swe cenne życie) postanowiłem jeszcze raz pokazać światu swoje stalowe nerwy i włączyłem zasilanie...

SŁÓW OGÓLNYCH KILKA

Rozszerzenie pamięci A504 PLUS jest przeznaczone dla Amigi 500 Plus i z założenia ma dać komputerowi dodatkowy megabajt pamięci typu Chip RAM. Tak tłumaczy nam dołączona do karty instrukcja obsługi. Gdyby jednak ktoś z Państwa łudził się, że coś z niej skorzysta, to powiem tylko, że drobną przeszkodą może być użyty tu język niemiecki. Zastanawiałem się bardzo, co by się stało, gdyby rozszerzenie to wstawić do Amigi 500 (z instrukcji jako żywo nie wynikało jasno, do jakiego komputera należy przyłączyć rozszerzenie), nikt jednak z moich znajomych komputer ten posiadających nie chciał się zgodzić na taki eksperyment.

INSTALACJA

Aby zainstalować opisywane rozszerzenie należy:
1. Wyłączyć komputer.
2. Zdjąć klapkę zakrywającą port rozszerzeń znajdujący się na spodzie Amigi.
3. Wsunąć w gniazdo płytkę rozszerzenia tak, aby układami scalonymi zwrócona była w kierunku klawiatury.



4. Założyć z powrotem klapkę.
5. Włączyć komputer.

Jeżeli instalacja przebiegła prawidłowo, to można dalej używać komputer wyposażony w 2 MB Chip RAM. Jeżeli nie, to należy:
1. Wyłączyć komputer.
2. Zanieść Amigę do najbliższego serwisu.

TESTY

Rozszerzoną pamięć sprawdziłem wszelkimi dostępnymi mi programami testującymi. Nie wykazała ona żadnych odchyleń od normy. Pamięć została rozpoznana przez system i w pełni zaakceptowana.

W DZIAŁANIU

Do obecnych teraz dwóch megabajtów pamięci Chip postanowiłem dodać jeszcze dwa megabajty pamięci typu Fast – wykorzystałem do tego celu rozszerzenie MEGAMIX (test tego rozszerzenia zamieszczony był w C&A 9/92). Wymiennie z nim podłączałem także polskie rozszerzenie pamięci MEGA RAM. W obydwu przypadkach system rozpoznał konfigurację pamięci w postaci 4 MB (2 MB Chip i 2 MB Fast).

Ponieważ doszedłem do wniosku, że same cyferki na belce Workbench'a jeszcze o niczym nie świadczą, postanowiłem porządnie sobie poszaleć. Pracowałem na takich programach jak Vista Pro 2 (wymagane minimum 3 MB pamięci), Cinemorph (im więcej, tym lepiej), Art Departament (minimum 2,5 MB pamięci), Scenery Animator (minimum 1,5 MB pamięci) i wielu innych. Podczas działania tych programów Amiga NIE ZAWIESIŁA SIĘ ANI RAZU.

W momencie gdy piszę ten artykuł, w pamięci znajdują się pracujące programy: CygnusED Professional, Vista Pro 2, Directory Opus v3.51, kilka podręcznych programów rezydentnych, AREXX i BaudBandit2.0 (program komunikacyjny, który od przeszło godziny próbuje wybrać numer jednego BBS-u). Dodatkowo do Amigi podłączony jest modem oraz dysk twardy (20 MB), na co zezwala przelotowość rozszerzenia MEGAMIX lub MEGA RAM. Wszystko działa bez najmniejszych prroblemów i nie ma mowy o jakimkolwiek zawieszaniu się systemu, pomimo że już kilkanaście godzin dostępna pamięć zajmowana jest w około 75-80 procentach, a procesor „zajęty jest po uszy" (generowanie grafiki w Vista Pro 2).

PODSUMOWANIE

Po kilkudniowych testach (w moim przypadku to normalna praca) rozszerzenie nie zawiodło ani razu. Śmiało mógłbym polecić je wszystkim posiadaczom Amigi 500 Plus, którzy posiadają tylko 1 MB pamięci Chip RAM. Urządzenie zostało sprawdzone i nie zagraża zdrowiu lub życiu użytkownika, a także nie powoduje uszkodzenia Amigi.

Donosi o tym z radością

Wasz Ryzykant

RAFAŁ BORZYŃSKI (RABOCOST)

Od redakcji:

Z firmą 3-STATE POLAND nawiązaliśmy bliższe kontakty, w związku z czym niebawem opublikujemy testy innych urządzeń pochodzących z bogatej oferty tejże firmy (m.in. twarde dyski + kontrolery do Amigi, rozszerzenia pamięci do A500, A500+ i A600, stacje dysków 3,5"). Zapraszamy więc do lektury C&A!

ZALETY:
+ rozszerzenie działa prawidłowo (z praktyki wiem, że trafić na feleme rozszerzenie wcale nie jest tak trudno, jakby się komuś wydawało)

WADY:
- brak instrukcji w języku polskim

CENA: 799.000 zł

DYSTRYBUTOR:
3-STATE POLAND,
ul. Orzycka 20, 02-695 Warszawa.
Telefony:
Warszawa - 43-16-56
Rybnik - 225-69
Radom - 417-42

AMIGA 600HD

Przez pierwsze kilka lat swego istnienia Amiga właściwie się nie zmieniała. Była wolna (jak na dzisiejsze czasy), nie miała twardego dysku. Dwa lata temu firma Commodore wypuściła na rynek Amigę 3000, mającą być komputerem dla ludzi potrzebujących Amigi do pracy – A500 i A2000 miały zaś stać się komputerami czysto domowymi. Nieco podobna sytuacja jest teraz, gdy od siedmiu lat nie zmieniają się możliwości graficzne Amigi - Commodore ponownie podejmuje próbę zróżnicowania rynku Amigi, zaś dwie skrajności mają reprezentować Amigi 600 i 4000.

O Amidze 600 mówi się już od dawna. Przez jednych uważana jest za dobre posunięcie, przez innych – za totalny niewypał. Atmosferę podgrzała w dodatku informacja, jakoby Commodore miało zupełnie zaprzestać produkcji Amigi 500 i skoncentrować się tylko na A600. Ucieszyłem się więc, gdy Amiga 600 przypadła mi do testowania – męczyłem ją kilka tygodni, teraz zaś przyszedł czas, bym podzielił się z Czytelnikami swoimi wrażeniami.

OD POCZĄTKU

Od początku, a więc od etapu rozpakowywania. Pudło A600 jest równie małe, jak ona sama. Oprócz komputera mieszczą się w nim zasilacz, myszka, cztery dyskietki i trzy książki, w tym jedna dość sporych rozmiarów. Dyskietki to Workbench 2.0, Extras 2.0, Fonts oraz A600HD Install.

Na dwóch pierwszych znajduje się pełny zestaw plików systemowych, począwszy od komendy "DIR" a skończywszy na programie do kompletnego konfigurowania twardego dysku. Trzeci dysk, Fonts, już od samego pojawienia się Workbench'a w wersji 2.04 stanowi dla wielu zagadkę. Otóż zawiera on trzy wektorowe czcionki w standardzie AGFA Compugraphic - Times, Triumvirate oraz LetterGothic. Są one tam, bowiem system w wersji 2.0 został przystosowany do współpracy z w pełni skalowalnymi, wektorowymi fontami. Zresztą przy użyciu dostarczanego z Amigą programu „Fountain" można wygenerować sobie wersje bitmapowe, bowiem na zwykłej, nie przyspieszonej Amidze czas dostępu do czcionek wektorowych jest potwornie długi.



Pozostał jeszcze czwarty dysk. Zawarte na nim programy służą tylko do jednego – przy ich pomocy możemy dokonać pełnej konfiguracji i instalacji twardego dysku. Na tym ich rola się kończy. Na „twardysku" A600 jest już firmowo instalowane oprogramowanie systemowe, i dla niektórych InstallDisk może wydać się zupełnie zbędny. Może on jednak przydać się, gdy zechcemy wymienić pliki systemowe na nowsze, na przykład gotowy już Workbench 2.1.

Jak wcześniej wspomniałem, z Amigą 600 otrzymałem trzy książki. Dwie z nich były w jakimś dziwnym języku (chyba po szwedzku), trzecia – po angielsku. Pierwsza książka zawiera absolutne podstawy – co gdzie się podłącza, gdzie szukać na klawiaturze klawisza RETURN itd. Druga wyjaśnia zasady pracy z Amigą – jak poruszać się po systemie, jak obsługiwać twardy dysk. Natomiast trzecia, ta po angielsku, zawiera szczegółową instrukcję pracy z systemem operacyjnym i może się bardzo przydać. Wszystko jest w niej wyjaśnione przejrzyście i dokładnie, a w dodatku zilustrowane dziesiątkami wydruków z ekranu.

Dołączoną literaturę oceniam bardzo wysoko, szkoda tylko, że nie jest ona w naszym ojczystym języku, a przynajmniej w całości w zrozumiałym dla wielu angielskim.

CO NOWEGO

Amiga 600 jest bardzo mała (na przykład od A500 jest aż o kilkanaście centymetrów węższa), wszystkie jej układy upakowano na niezwykle małej płycie głównej. Za ową miniaturyzację trzeba jednak zapłacić: otóż tylko kość zawierająca ROM jest umieszczona w podstawce. WSZYSTKIE pozostałe kości lutowane są techniką montażu powierzchniowego. Korzyści z tego są takie, że płyta może być niezwykle mała, a koszty produkcji są bardzo niskie. Do wad należy fakt, że sprzęt tak wykonany jest praktycznie nienaprawialny. Po pierwsze kostki zastosowane w A600 muszą się – ze względu na technikę montażu – różnić od tych, które są montowane w zwykłych Amigach. Oznacza to, że 95% „punktów serwisowych" oświadczy od razu, że nie ma części. To jednak nie jest jeszcze największy problem. Zwykłą Amigę naprawia się (przynajmniej w Polsce) wyciągając po kolei układy scalone i wkładając na ich miejsca sprawdzone, na pewno dobre. Gdy zadziała – usterka jest wykryta. Nie da się tego manewru przeprowadzić przy serwisie A600 – trzeba mieć specjalizowany sprzęt, który przetestuje układy bez odrywania ich od płyty głównej. Odlutowanie układu, gdy ma się specjalną lutownicę, jest dość proste. Gorzej jest z przylutowaniem - to już praca dla zegarmistrza. Zresztą w praktyce rzadko zdarza się, by udało się przeprowadzić taką operację – mam na myśli odlutowanie układu i zamontowanie go z powrotem – więcej niż raz bez uszkodzenia jego lub ścieżek na płycie.

Na Zachodzie, gdzie serwis jest wyposażany i szkolony przez producenta, z pewnością będzie łatwiej o naprawę A600. Podejrzewam co prawda, że płytę główną A600 będzie się tam traktowało jako jednorazówkę i po prostu wymieniało w całości. W każdym razie w Polsce, przy dzisiejszym poziomie usług serwisowych, A600 po upływie gwarancji jest komputerem dla ryzykantów. Bardzo daleki jestem jednak od dyskwalifikowania jej z tego powodu – w końcu Amigi nie psują się zbytnio, a jeżeli już, to nie bez powodu.



No ale miałem pisać o tym, w co nowego wyposażono A600. Liczące się nowości są dwie. Pierwsza z nich to gniazdo krzemowych kart pamięci PCMCIA. Jak na razie są one jeszcze wynalazkiem znanym dla wielu tylko ze słyszenia, warto więc chyba poświęcić im kilka linijek.

Karta PCMCIA wielkością przypomina kartę kredytową, jest jednak nieco grubsza. Zapisywane na niej dane nie są tracone po wyłączeniu zasilania komputera, zaś aktualnie wytwarzane karty mają pojemność do czterech megabajtów. Wydaje się, że karty te mogłyby zyskać sporą popularność, choćby z uwagi na ich doskonałe walory użytkowe (duża szybkość transmisji, małe rozmiary, przechowywanie danych po odłączeniu od komputera), jednak ich cena jest jak na razie bardzo wysoka - użyta przeze mnie do testów karta o pojemności 512 KB kosztowała prawie 300 dolarów.

Podłączona do A600 karta PCMCIA może pracować jako bardzo szybka stacja dysków, lub też symulować pamięć FAST. O ile mi wiadomo, użycie karty PCMCIA jest na razie jedynym sposobem uzyskania w Amidze 600 pamięci FAST, bowiem gniazdo pod klawiaturą z założenia służy do rozbudowy CHIP-RAM.

Warto tu jeszcze wspomnieć o jednej ciekawej rzeczy. Użytą przy testach kartę pożyczyłem od szefa, używającego jej do swego palmtopa firmy Hewlett Packard. Komputer ten stosuje takie karty jako coś w rodzaju mini-twardego dysku, przy czym format zapisu jest zgodny z MS-DOS. Otóż, z wykorzystaniem programu MultiDos, udało mi się odczytać na Amidze 600 dane zapisane przez palmtopa. Nie było również kłopotów z zapisem. Tak więc przy pomocy A600 można swobodnie (bez uciekania się do transmisji przez RS-232) przenosić dane pomiędzy palmtopem a Amigą (lub też Atari ST i IBM PC, których formaty Amiga „rozumie").

Druga ze wspomnianych nowości to sterownik dysków twardych AT-BUS. Możliwa jest dzięki niemu instalacja w A600 napędu dysku twardego w tym standardzie. Problem jednak w tym, że nie przewidziano podłączenia go z zewnątrz, zaś wewnątrz A600 jest tak mało miejsca, że konieczne jest kupno dysku 2,5-calowego, czyli bardzo małego i, co za tym idzie, niezbyt taniego.

W testowanym modelu A600HD, jak sama nazwa wskazuje, twardy dysk jest montowany fabrycznie. Poza tym, że wewnątrz ma napęd AT-BUS, Amiga 600HD NICZYM nie różni się od zwykłej A600. Warto tu jeszcze wspomnieć o niespodziance, jaką przyniosło pierwsze uważne przyjrzenie się ilości wolnego miejsca na dysku. Otóż, wbrew zapowiedzi na pudle, mówiącej o „20 MB HDD", okazało się, że zamontowany napęd ma pojemność 30 MB.

Nieco mniej znaczącą nowością jest fabrycznie montowany w Amidze 600 modulator RF, dzięki któremu można na każdym telewizorze pracującym w systemie PAL odbierać obraz i dźwięk z Amigi.

Poza gniazdem PCMCIA, sterownikiem AT-BUS i modulatorem w Amidze 600 nie ma absolutnie nic nowego. Procesory specjalizowane mają dokładnie identyczne parametry jak te w Amigach 500+ i 3000. Tak więc graficzne i muzyczne możliwości Amigi 600 pozostały nie zmienione. Nie ulepszono także sterownika stacji dysków, a co za tym idzie, nadal nie ma możliwości zapisania na dyskietce więcej niż 880 KB.

Firma Commodore poszła po prostu na łatwiznę - dodanie złącza PCMCIA i śmiesznie prostego interfejsu AT-BUS trudno bowiem zaliczyć do zmian rewolucyjnych. Amiga 600 jest komputerem czysto domowym, nie zmienia to jednak faktu, że od czasu, gdy jej możliwości graficzne i muzyczne były rewelacyjne, minęło sporo lat i dziś nawet Atari (model Falcon) bije A600 na głowę.

PRAKTYKA

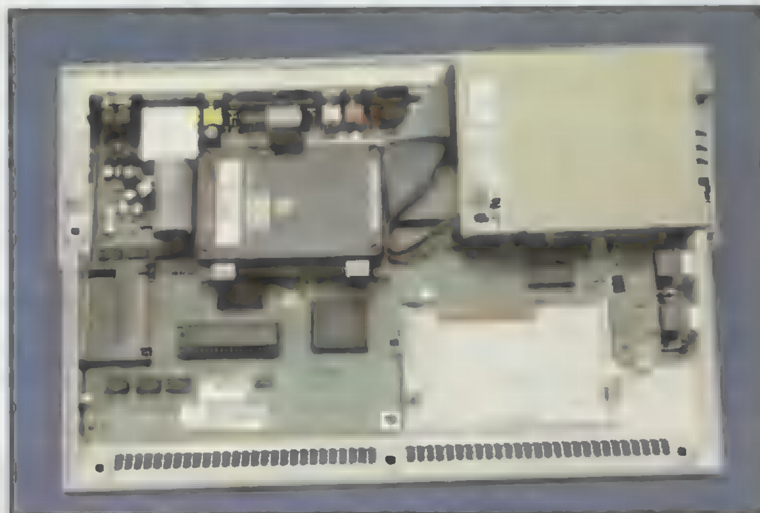
Wypada chyba zacząć od idealnej zgodności programowej A600 z A500+. Wynika to z faktu, że wszystkie elementy ją determinujące są w obu komputerach takie same. Niczym się nie różnią ich kości specjalizowane, dotyczy to również systemu operacyjnego. Wersja instalowana w Amidze 600 (2.05) ma co prawda nieco inne oznaczenie niż ta znana z A500+ (2.04), w praktyce jednak nie da się zauważyć jakichkolwiek różnic pomiędzy nimi.

Wśród nowych programów użytkowych nie natrafiłem na ani jeden nie działający na A600. Jeżeli były takie wśród starych, to już dawno wypuszczono poprawione wersje. Z pełną odpowiedzialnością twierdzę, że w odniesieniu do programów użytkowych problem niekompatybilności Amigi 600 praktycznie nie istnieje. Minimalnie gorzej jest z grami i demami – z nowości około 5% odmawia współpracy. Trzeba jednak pamiętać o starszym oprogramowaniu. Im starsza gra czy demo, tym większe prawdopodobieństwo, że Kickstart 2.0 mu się nie spodoba. Nie ma na to rady – nie widać jak na razie ofert instalacji dodatkowego ROM'u w Amidze 600. Wszystko wskazuje jednak na to, że pojawią się one niebawem.

Warto też wspomnieć o twardym dysku. Trzeba pamiętać, że twardy dysk to napęd oraz sterownik, na który składa się część sprzętowa i program sterujący. Zdarza się, że, mówiąc ogólnie, wszystkie te elementy nie współpracują ze sobą idealnie. Objawiać się to może na przykład nieuzasadnionymi błędami tzw. walidacji dysku (uaktualniania mapy jego zajętości) lub po prostu tajemniczymi uszkodzeniami plików. Zdarza się też, że poprawnie na codzień pracujący „twardysk" żadnym sposobem nie chce się poddać optymalizacji. Na szczęście z dyskiem Amigi 600 nie było żadnych tego rodzaju kłopotów. Jako ciekawostkę mogę natomiast dodać, że program sterujący nim nazywa się ni mniej, ni więcej, tylko „scsi.device". Przypominam, że napęd zainstalowany w A600 jest typu AT-BUS. Niestety, z powodu braku dodatkowego złącza, nie jest możliwe podłączenie więcej niż jednego napędu.

Odmienną natomiast sprawą są parametry „twardysku" Amigi 600. Szybkość transmisji raczej nie jest rewelacyjna. Dla porównania dodam, że bardzo przeciętny dysk twardy Supra osiąga prędkość trzy do pięciu razy większą, zaś jeden z lepszych, produkcji firmy GVP, w skrajnych przypadkach nawet dziesięć razy większą. Nie można jednak zapominać o tym, że zastosowany w A600 napęd należy do najmniejszych (mam oczywiście na myśli rozmiary), a przecież nic nie ma za darmo.

Ostatnia sprawa to możliwości rozbudowy. Nie są one olbrzymie. Pod klawiaturą, podobnie jak w A500, umieszczono złącze rozszerzenia pamięci. Wymaga ono specjalnego modułu, przy pomocy którego można rozbudować pamięć CHIP Amigi 600 do 2 MB. Nic jak na razie nie wiadomo o jakichkolwiek innych modułach rozbudowujących możliwości A600. Co więcej, w komputerze tym nie ma szyny procesora, z której (w przypadku A500) korzysta 99% dopalaczy, dużych rozszerzeń pamięci, twardych dysków itd.



Tak prezentuje się A600 od środka

PODSUMOWANIE

Tekst ten piętnuje wiele wad Amigi 600 i niektórzy z Czytelników spodziewają się zapewne, że w tej chwili zdecydowanie odradzę kupno tego komputera. Otóż nie, Amiga 600 nie jest bowiem przeznaczona dla fanów grafiki czy w ogóle ludzi używających komputera do poważniejszych celów. Jest ona maszyną DOMOWĄ, której najczęstszym zadaniem jest dostarczanie rozrywki dzieciom, czasem zaś może ona ułatwić głowie rodziny prowadzenie rachunków domowych czy też napisanie tekstu. Gdy spojrzymy na A600 w ten sposób, dojdziemy do wniosku, że nie jest ona takim złym posunięciem firmy Commodore. Jest bardzo mała, łatwa w podłączeniu i obsłudze, a przy tym oferuje niezłe możliwości graficzne i muzyczne, których jedynym zastosowaniem nie musi być uatrakcyjnianie mordowania kosmitów - dla Amigi napisano przecież wiele wartościowych programów edukacyjnych, które z powodzeniem będą pracować na A600.

ANDRZEJ BOBEK

ZALETY:
+ małe rozmiary
+ dość pożyteczne usprawnienia
+ wbudowany twardy dysk

WADY:
– niemal zerowe możliwości rozbudowy
– bardzo utrudniony serwis
– niezmienione możliwości graficzne i muzyczne



DANE TECHNICZNE
Procesor: MC68000, 7,09 MHz (PAL)
lub 7,16 MHz (NTSC)
Pamięć: CHIP – standardowo 1 MB, rozszerzalna do 2 MB
SLOW – niedostępna
FAST – standardowo brak, przez złącze PCMCIA rozszerzalna nawet do 4 MB
ROM – 512 KB, Kickstart 2.05
Tryby graficzne:
– LoRes/LoRes Interlaced
320*256/512 – 2, 4, 8, 16, 32, 64 lub 4069 kolorów
– HiRes/HiRes Interlaced
640*256/512 – 2, 4 lub 16 kolorów
– Productivity
640*480 – 2 lub 4 kolory, sygnał 31 kHz (non-interlaced, standard VGA)
– Productivity Interlaced
640*960 – 2 lub 4 kolory
– SuperHires/SuperHires Interlaced
1280*256/512 – 2 lub 4 kolory
Dźwięk: stereo, 4 niezależne 8-bitowe przetworniki cyfrowo-analogowe, po dwa na każdy kanał stereo; 8 oktaw
Stacja dysków: dwustronna, 880 KB
Twardy dysk:
– rozmiar talerza: 2,5 cala
– pojemność: 20 MB (w testowanym egzemplarzu; wg Commodore do Amig 600HD będą montowane dyski twarde o pojemnościach od 20 do 120 MB)
– typ interfejsu: AT-Bus
Klawiatura: 76 klawiszy, brak bloku numerycznego
Możliwości rozbudowy:
– brak złącza procesora
– złącze rozszerzenia pamięci – umożliwia rozszerzenie CHIP do 2 MB przy pomocy specjalnego modułu
Złącza:
– 2 * mysz/joystick (standardowe, dziewięciostykowe)
– dodatkowych stacji dysków (do trzech)
– szeregowe (RS-232)
– równoległe (Centronics)
– monitora RGB
– composite video
– telewizora (modulator RF)
– kart pamięci PCMCIA
– rozszerzenia pamięci (pod klawiaturą)
– dwa złącza dźwięku stereo

DYSTRYBUTOR: **JTT COMPUTER**
Wrocław, ul. Świdnicka 19
tel. (071) 44 12 33
fax (071) 44 66 69



Tylne gniazda A 600.
Złącze umożliwiające podłączenie kart PCMCIA znajduje się z lewej strony komputera.

PORADNIK POCZĄTKUJĄCEGO AMIGANTA (cz. 1)

REMANENT

Przygotowując cykl artykułów o AmigaDOS pomyślałem, że warto by było zrobić remanent, ponieważ temat ten przewijał się już na naszych łamach. Nie będę powtarzał informacji zawartych w opublikowanych artykułach, ale je tylko uzupełnię o moim zdaniem ważne dla początkującego użytkownika sprawy.

PODSTAWY PODSTAW

Zacznę od katalogów. AmigaDOS pozwala na budowanie struktury dyskietki w formie drzewa. Jego podstawą jest katalog główny (root), w którym umieszczone są podkatalogi (między innymi systemowe, takie jak c, devs, i itd.). W nich z kolei mogą być inne podkatalogi zawierające jeszcze inne itd. Przykładowa struktura dyskietki jest pokazana na rysunku.

Do zmiany katalogu aktualnego służy komenda CD (Change Directory). Załóżmy, że jesteśmy w katalogu głównym. Aby się znaleźć w katalogu Archiwizery, trzeba wpisać:
CD Użytki/Archiwizery
Ale aby przejść z powrotem do katalogu głównego, wystarczy wprowadzić:
CD // lub CD :

Jeden znak / (slash) powoduje przejście do katalogu o jeden wyższego, więc dwa w tym przypadku skierują nas do katalogu głównego. Natomiast dwukropek zawsze skieruje nas do katalogu głównego.

Bardziej skomplikowane jest przejście np. z katalogu "Archiwizery" do "Listy" - aby to zrobić należy wpisać:
CD :Teksty/Listy
Po tym przykładzie przejście z katologu Listy do Artykuły nie powinno sprawiać kłopotów. Piszemy więc:
CD /Artykuły
i jesteśmy tam, gdzie chcieliśmy.

W przypadku, gdy wędrując po katalogach stracimy orientację, możemy wpisać samo CD, wtedy system poinformuje nas, gdzie się znajdujemy. Innym sposobem na zorientowanie się gdzie jesteśmy, jest wykorzystanie komendy PROMPT. Można w jej parametrach wpisać dwa znaki specjalne:
„%N" - będziemy informowani o numerze procesu (a mówiąc po ludzku: numerze CLI lub Shell) na jakim pracujemy,
„%S" - będzie podawana pełna ścieżka od katalogu głównego do aktualnego, a dodatkowo można umieścić ciąg tekstowy, jest to tzw. znak zachęty. Na przykład:
PROMPT „%N.%S> Czego dusza chciała ?>"
spowoduje pojawienie się znaku zachęty w postaci:
1.Mój_dysk:Teksty/Artykuły> Czego dusza chciała ?>
gdzie 1 oznacza numer procesu, a „Mój_dysk..." to nazwa dysku („Mój dysk:") i tzw. ścieżki dostępu, czyli katalogu nas interesującego (Artykuły) i innych, leżących niejako po drodze (Teksty).

SPOSOBY...

Wpisywanie pełnych nazw ścieżek jest trochę uciążliwe, a jest to niezbędne w przypadku uruchamiania programów z katalogów innych, niż aktualny lub C:. Pomocną w tym przypadku będzie komenda PATH. Pozwoli ona podpowiedzieć systemowi, gdzie ma szukać programu w przypadku, gdy nie podamy ścieżki dostępu. Na przykład chcemy rozpakować archiwum „Fish.lha" z katalogu „Archiwum" programem Lha z katalogu Archiwizery do RAM'u. W skrajnym przypadku, kiedy katalogiem aktualnym nie jest ani katalog Archiwum, ani Archiwizery, ale np. główny, piszemy:
Użytki/Archiwizery/Lha e Archiwum/Fish.lha ram:
Sytuacja zmienia się, gdy wcześniej wpiszemy:
PATH Mój_dysk:Użytki/Archiwizery ADD
Teraz niezależnie od tego, w którym katalogu będziemy, system sam znajdzie program Lha. Dodatkowo przejście komendą CD do katalogu "Archiwum" pozwoli skrócić polecenie do minimum:
Lha e Fish.lha ram:

Podanie komendy PATH bez parametrów wyświetli listę katalogów według kolejności ich przeszukiwania przez system. Na kolejność tą można wpływać podając w odpowiedni sposób katalogi do przeszukania, przy wykonaniu komendy np.:
PATH ram: sys: sys:c sys:Użytki sys:Użytki/Archiwizery
spowoduje, że system będzie przeszukiwał katalogi w takiej kolejności:
1. katalog aktualny (zawsze jako pierwszy)
2. ram:
3. główny (dyskietki systemowej)
4. sys:c (czyli c:)
itd., a ostatnim będzie znowu c:.
Opcja RESET skasuje całą listę. Wtedy przeszukiwany będzie tylko katalog aktualny i „c:".

...I SPOSOBIKI

Niestety, nie we wszystkich przypadkach to pomaga. Jeśli na przykład nazwę pliku musimy podać jako parametr dla innego programu, to nadal trzeba podać pełną ścieżkę dostępu, jeżeli nie ma go w katalogu aktualnym. Możemy sobie ułatwić życie wykorzystując komendę ASSIGN. Zamiast wypisywać tasiemcowe ścieżki, stwórzmy dwa urządzenia logiczne:
ASSIGN Text1: Mój_dysk:Texty/Listy
oraz
ASSIGN Text2: Mój_dysk:Texty/Artykuły
Teraz każde odwołanie do urządzenia Text1: i Te-



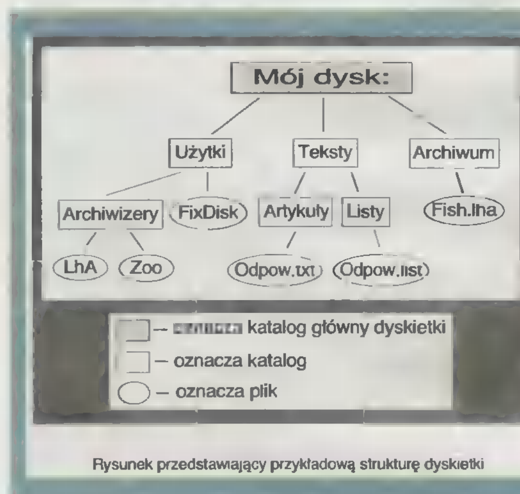
xt2: będzie interpretowane przez system tak samo, jak do odpowiedniego katalogu. Np.:
JOIN Text1:Odpow.list Text2:Odpow.txt TO ram:Nowy.txt
połączy plik „Odpow.list" z katalogu Artykuły z plikiem „Odpow.txt" z katalogu Listy, całość zapisze do RAM'u pod nazwą „Nowy.txt". Dodatkową zaletą tego sposobu jest to, że polecenie zostanie wykonane niezależnie od tego, w którym katalogu obecnie się znajdujemy. Oczywiście pod warunkiem, że JOIN będzie w katalogu „c:". Należy tu zwrócić uwagę, iż nazwy urządzeń logicznych kończą się dwukropkiem.

NIBY NIC WIELKIEGO, ALE...

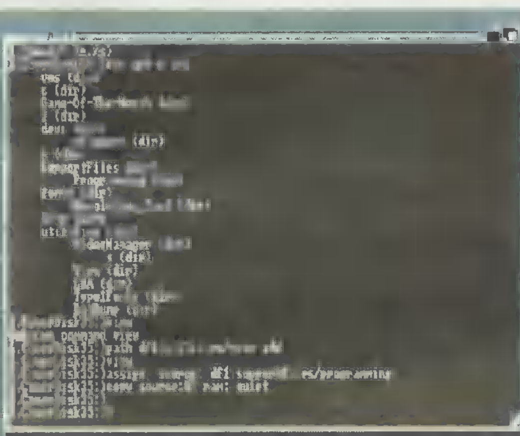
W ostatnim przykładzie w nazwach programów zastosowałem rozszerzenia (znaki po kropce). Używanie ich pozwala bardzo szybko zorientować się, z jakim rodzajem pliku mamy do czynienia i ułatwia przeprowadzenie tej samej operacji na wielu plikach jednocześnie. Bardzo pomocne w tym wypadku są znaki uniwersalne (ang. wildcard characters). Jakie znaki rozpoznaje AmigaDOS, ilustruje tabela 1. Kilka przykładów pozwoli zorientować się, o co w tym wszystkim „chodzi".
COPY df0:#? ram:
skopiuje wszystkie pliki z df0: do ram: (ale tylko z katalogu głównego - aby skopiować całość, łącznie z podkatalogami i ich zawartością, należy dodać opcję ALL).
DELETE #?.(bas | bak)
skasuje wszystkie pliki z rozszerzeniem .bas i .bak z aktualnego katalogu.
TYPE a#st
wyświetli zawartość plików: ast, asst, assst itd.
DELETE proba?
skasuje wszystkie pliki o nazwie: proba1, proba2, itd. ale nie skasuje np. pliku proba10.

ACH TEN KAPRYŚNY AMIGADOS

Uzupełniając sposoby „radzenia sobie" z AmigaDOS (C&A 5/92) należy wspomnieć o komendzie RESIDENT. Dzięki niej możemy wprowadzić inne komendy systemowe do pamięci. Po tej operacji system nie będzie potrzebował dysku systemowego do ich wczytania, bowiem będzie je miał „pod ręką". Jaka jest różnica, można się przekonać na przykładzie dwóch podobnych w działaniu komend DIR i LIST. Startujemy „z Workbench'a", czekamy aż wykona się cała (oryginalna!) sekwencja startowa i uruchamiamy Shell. Teraz wydajmy polecenie:
DIR ram:
System zacznie czytać polecenie DIR z dyskietki i po chwili wyświetli zawartość RAM'u. Następne wydajmy polecenie:
LIST ram:
Dla większego efektu wyjmijmy wcześniej dyskietkę ze stacji. Mimo braku dyskietki systemowej w stacji komenda wykonała się poprawnie. A to dzięki temu, że gdzieś w sekwencji startowej została wprowadzona do pamięci komendą RESIDENT. Sami też możemy to zrobić odnośnie innych komend:
RESIDENT Dir ADD
spowoduje, że teraz i dir będzie poleceniem rezydentnym.
Niestety nie wszystkie komendy i programy można uczynić rezydentnymi. Na pewno można to zrobić ze wszystkimi komendami z katalogu „c" Workbench'a, które mają ustawiony bit p (pure). Tu uwaga: niektóre programy kopiujące kasują ten bit przy kopiowaniu. Należy go zatem ustawić poleceniem:
PROTECT nazwa +p.
W innym przypadku otrzymamy ostrzeżenie „Pure bit not set"(bit p nie jest ustawiony) i komenda się nie zainstaluje, chyba że zastosujemy opcję PURE:
RESIDENT Echo PURE ADD
Teraz komenda zainstaluje niezależnie od tego, czy jest ustawiony bit p, czy nie (ale ostrzeżenie nadal będzie wyświetlane). Listę poleceń rezydentnych otrzymamy wpisując polecenie RESIDENT bez parametrów. Natomiast
RESIDENT Echo REMOVE
usunie polecenie Echo z listy.
Z innymi programami należy postępować ostrożnie. Niektóre z nich mogą być rezydentne, inne nie, jedyną drogą do sprawdzenia tego jest eksperyment. Na koniec uwaga: metoda ta oprócz zalet ma i wady - każde zainstalowane polecenie lub program zmniejsza nam ilość wolnej pamięci.



Rysunek przedstawiający przykładową strukturę dyskietki


Tak właśnie wygląda CLI

CO NOWEGO POD 2.0

Wiele poleceń AmigaDOS'u jest już na stałe w pamięci. Nie zajmują tym samym miejsca na dyskietce. Które - można się dowiedzieć wpisując polecenie RESIDENT bez parametrów. Zmieniając katalog aktualny nie trzeba już podawać magicznego słowa CD. Nazwa katalogu lub symbol stacji wystarczy.
df0:
znaczy to samo co:
CD df0:
Użycie polecenia PROMPT bez parametrów powoduje wyświetlanie znaku zachęty w postaci domyślnej %N.%S. Pojawił się nowy znak uniwersalny ~ (falka) - jest to znak logicznego zaprzeczenia. I tak
DELETE ~(#?.bak)
skasuje wszystkie pliki nie mające rozszerzenia .bak.
Na koniec mała ciekawostka: jeżeli podczas uruchamiania systemu będziemy trzymać wciśnięte oba klawisze myszy, to na ekranie pojawi się menu, dzięki któremu możemy wybrać między innymi stację, z której system będzie startował ("bootował się") w pierwszej kolejności, oraz czy ma się wykonać Startup-Sequence, czy też nie. Menu to daje nam jeszcze jedną ciekawą możliwość: możemy powyłączać dowolnie wybrane urządzenia, co oznacza na przykład, że wyłączniki w zewnętrznych stacjach dysków są posiadaczom systemu 2.0 zbędne.
To by było na tyle w tym miesiącu. ENDCLI.

PAWEŁ GALAS

W następnym miesiącu: Startup-Sequence

Tabela 1

ZNAKI UNIWERSALNE	
?	– zastępuje jeden dowolny znak
#	– pozwala zastąpić wiele jednakowych znaków jednym
|	– (logiczne LUB) używane do zastępowania znaków więcej niż jednym wzorcem
(i)	– służą do grupowania wzorców
%	– zastępuje pusty ciąg
Najczęściej używanym zestawieniem jest: #? – zastępujący każdy ciąg znaków.	

Polecenia poznane w tym miesiącu

ASSIGN – przypisuje katalogowi urządzenie logiczne

CD katalog – zmienia katalog aktualny

COPY – wykonuje kopię pliku w innym katalogu lub na innej dyskietce

DELETE – kasuje (niszczy!) plik z dyskietki

DIR – wyświetla zawartość katalogu

JOIN – łączy pliki tekstowe w jedną całość

LIST – wyświetla zawartość katalogu z podaniem wielkości plików, czasu utworzenia, bitów protekcji i komentarza

PATH – dodaje katalog do listy katalogów przeszukiwanych przez system

PROMPT – zmienia tzw. znak zachęty

PROTECT – ustawia lub kasuje bity protekcji

RESIDENT – wpisuje polecenie do listy poleceń rezydentnych

TYPE – wyświetla zawartość pliku tekstowego

PROCEDURY

W poprzednim odcinku kursu poznaliśmy kombinację instrukcji GOSUB-RETURN, które tworzą podprogramy. Dziś poznamy procedury. W swojej zasadzie działania są one podobne: program wykonuje się do momentu wywołania procedury, wykonuje skok do niej, po zakończeniu wraca do pierwszej instrukcji umieszczonej po wywołaniu procedury. Zasadnicza różnica między procedurami, a podprogramami wywołanymi przez GOSUB, to obecność zmiennych lokalnych. Krótko mowiąc procedura działa jak oddzielny program: modyfikowanie zmiennych wewnątrz nie spowoduje ich zmiany w programie zasadniczym.

W takim układzie można sobie zadać pytanie, jak odbywa się komunikacja takiego segmentu z resztą programu. Może się to odbywać za pomocą zmiennych, które zdefiniujemy jako globalne, tzn. będą one wspólne dla programu i wszystkich procedur, bądź też za pomocą parametrów. A teraz już do rzeczy.

AMOS cz. IV

Nazwa procedury to ciąg znaków ujętych takimi prawidłami, jak przy tworzeniu etykiet. Procedurę rozpoczynamy w następujący sposób:

```
Procedure CZESC
```

gdzie CZESC jest nazwą procedury. Kończymy ją instrukcją

```
End Proc
```

Zobaczmy teraz na przykładzie jak to działa:

```
CZESC
Print A
'
Procedure CZESC
  Print „Witam wszystkich serdecznie !!!"
  A=10
End Proc
```

Po wykonaniu na ekranie ujrzymy:
Witam wszystkich serdecznie !!!
0

Zmienna A wewnątrz procedury nie jest tą samą zmienną co na zewnątrz. Żeby nią była, należy użyć instrukcji Global:

```
Global A
CZESC
Print A
'
Procedure CZESC
  Print „Witam wszystkich serdecznie !!!"
  A=10
End Proc
```

Wtedy wydruk będzie miał postać:
Witam wszystkich serdecznie !!!
10

Czy jest to zrozumiałe? Zmiennych podanych w instrukcji GLOBAL można podać więcej, należy je wtedy rozdzielić przecinkami. Mogą one być modyfikowane wewnątrz każdej wykonywanej procedury. Innym sposobem komunikacji jest instrukcja SHARED. Jest to jednak komunikacja jednostronna, tzn. tylko z zewnątrz do wewnątrz procedury:

```
A=10
CZESC
'
Procedure CZESC
  Shared A
  Print „Witam wszystkich serdecznie !!!"
  Print A
End Proc
```

Instrukcja SHARED powoduje pobranie z „zewnątrz" wartości zmiennych podanych w jej parametrze (może ich być więcej, oddzielamy je wtedy przecinkami).

Trzecim sposobem są procedury wywoływane z parametrami, np

```
KOLO[100,100,50]
'
Procedure KOLO[X,Y,R]
  Circle X,Y,R
  Paint X,Y
End Proc
```

W AMOSie nie ma instrukcji rysującej koło. Musimy zastąpić ją procedurą składającą się z instrukcji rysującej okrąg - CIRCLE i wypełniającej go - PAINT (oczywiście będziemy o nich mówić dokładnie). Parametry pobierane przez procedurę to:
X - współrzędna pozioma środka koła
Y - współrzędna pionowa środka koła
R - promień koła
Umieszcza się je po przecinku w nawiasie kwadratowym. Jest to bardzo proste, wobec czego nie będę tego dokładniej omawiać.

Pozostaje jeszcze sprawa wyjścia z procedury. Czy można przekazywać w taki sposób parametry na zewnątrz? Tak, popatrzmy na przykład:

```
Print „Podaj czy chcesz dalej dalej kontynuowac (t/n)"
TAKNIE
If Param=1
  Goto POCZATEK
Else
  Print „To czesc !!!"
  End
End If
'
Procedure TAKNIE
  POCZATEK:
  T$=Lower$(Inkey$)
  If(T$<>"t") and(T$<>"n")
    Goto POCZATEK
  End If
  If T$="t"
    TAK=1
  Else
    TAK=0
  End If
End Proc[TAK]
```

Procedura TAKNIE służy do sprawdzenia, czy użytkownik naciśnie klawisz „t" czy „n". Jeżeli naciśnie „t", to na zewnątrz procedury zostanie oddana wartość 1, jeżeli „n" - to 0. Dzieje się tak za sprawą zmiennej TAK, która jest umieszczona w zakończeniu procedury: End Proc[TAK]. Jeżeli teraz chcemy ją odczytać, to musimy skorzystać ze specjalnej zmiennej Param. Ona właśnie kryje wartość zmiennej TAK. Jeżeli zmienna TAK byłaby zmienną rzeczywistą (patrz odcinek kursu o typach zmiennych), to aby odczytać jej wartość należałoby użyć zmiennej Param#, a jeśli tekstową - Param$.

Do tego tematu należy jeszcze dodać jedną ważną instrukcję. Służy ona do opuszczenia procedury w innym miejscu niż na jej końcu. Służy do tego instrukcja
Pop proc
Zobaczmy na przykładzie:

```
Print „Poczatek programu"
Print
DEMO
Print
Print „Koniec programu"
'
Procedure DEMO
Print „Czesc, jestes teraz wewnatrz procedury"
Print „Chcesz kontynuowac? Jezeli tak,"
  Print „to nacisnij 't', a jezeli nie to"
  Print „Nacisnij 'n' i za pomoca 'Pop Proc'"
  Print „zakonczysz program."
  TAKNIE
  If Param=0
    Pop Proc
  End If
  Print „Gratuluje Ci, ze zdecydowales sie"
  Print „Na dalsza wspolprace, ale"
  Print „niestety to juz koniec programu"
End Proc
```

```
Procedure TAKNIE
  POCZATEK:
  T$=Lower$(Inkey$)
  If(T$<>"t") and(T$<>"n")
    Goto POCZATEK
  End If
  If T$="t"
    TAK=1
  Else
    TAK=0
  End If
End Proc[TAK]
```

Widzimy więc także, że z wewnątrz jednej procedury można wywołać następną. Spod procedury DEMO została wywołana procedura TAKNIE wykorzystana już w poprzednim programie.

Jeżeli coś jest niezrozumiałe, proponuję przy uruchamianiu powyższych programów używać instrukcji FOLLOW, omówionej w poprzednim odcinku. Pozwoli ona dokładnie prześledzić wykonywane przez komputer operacje.

GRAFIKA

Grafika Amigi to krótko mówiąc *screeny* i okna. AMOS, pomimo swej odmienności od standardu środowiska graficznego, także oparty jest na strukturze tych dwóch elementów. Gdy zaraz po wczytaniu interpretera naciśniemy klawisz Esc, ujrzymy „obszar roboczy AMOSa". Ten pomarańczowy prostokąt na ekranie monitora to po prostu otwarty *screen* w niskiej rozdzielczości, a na nim na całej jego powierzchni otwarte jest okno numer zero (podczas otwarcia *screenu* okno to jest otwierane automatycznie, co pozwala dokonywać na nim operacji graficznych i tekstowych; nie można tego robić na screenie - spróbuj w jakimś innym języku, np. w Amiga BASIC'u). Nauczymy się teraz jak otwierać *screeny*.

SCREENY

Zamknijmy najpierw *screen*, który otwarty jest na początku. Służy do tego instrukcja:

Screen Close X

gdzie X jest numerem *screenu*. W naszym przypadku jest to 0, tak więc:

Screen Close 0 zamknie nasz podstawowy „pomarańczowy prostokąt". Nie wierzycie, sprawdźcie - po wykonaniu tej instrukcji ekran powinien stać się czarny. Teraz otwórzmy nowy. Służy do tego instrukcja:

Screen Open X,SZER,WYS,ILKOL,ROZD

A teraz znaczenie poszczególnych parametrów:

X - numer *screenu* - może być od 0 do 7
SZER - szerokość *screenu*
WYS - wysokość *screenu*
ILKOL - ilość dostępnych kolorów
ROZD - tryb rozdzielczości Amigi

Maksymalna ilość kolorów jakie chcemy wykorzystać jest zależna od trybu rozdzielczości Amigi. Co i jak można powłączać przedstawia tabelka:

[illegible]	[illegible]	[illegible]	[illegible]
LOWRES	[illegible]	[illegible] 64,4096	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]
LOWRES-LACE	[illegible]	[illegible] 64	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]

Kolumna „Rozmiar" zawiera optymalny rozmiar *screenu* dla NTSC, a w nawiasie dla PAL. W żadnej, jak już wspominałem dokumentacji, nie znalazłem informacji jak otrzymywać tryby LOWRES-LACE i HIRES-LACE. Tu podaję jak to robić. Jak więc widać, w AMOSIE dostępne są wszystkie tryby graficzne za wyjątkiem LOWRES-LACE przy 4096 kolorach. Przykładowo więc otwórzmy sobie *screen*:

```
Screen Close 0
Screen Open 0,640,512,16,-1
Print „Oto otwarty screen w najwyzszej rozdzielczosci,"
Print „w 16 kolorach"
```

Proponuję potrenować. Gdyby zdarzyły się jakieś problemy, wystarczy napisać list, na który odpowiem osobiście. Życzę wiele przyjemności, jakich dostarczyć może Wam AMOS. Do zobaczenia za miesiąc.

RAFAŁ BORZYŃSKI (RABOCOST)

W związku z dużą ilością poznawanych instrukcji postanowiłem - dla przypomnienia i porządku - dokonać krótkiego zestawienia i podsumowania. Wyrażenia ujęte w nawiasach klamrowych {} są opcjonalne. W dużej części opis tych instrukcji nie jest pełny, lecz tylko taki, jak je omawialiśmy. Z czasem będę uzupełniał informacje o pozostałych możliwościach podanych instrukcji.

Draw To - [illegible] X1,Y1 To X2,Y2 [illegible] X2,Y2 [illegible] rysowania [illegible] Draw 0,50 To 110,150

Print - [illegible] do wyprowadzenia na ekran [illegible] komunikat
[illegible] Print [illegible]
gdzie [illegible] typem zmiennej [illegible] lub tekstu [illegible] Przecinek umieszcza wyprowadzane informacje w [illegible] natomiast [illegible] za drugą
[illegible]

Input - [illegible]
[illegible] ZMIENNA
[illegible] łańcuch [illegible] linię podpowiedzi
ZMIENNA - zmienna, do której wpisana jest [illegible] od użytkownika [illegible]
[illegible] swój wiek [illegible]

If Then - [illegible] Pozwala uzależnić [illegible] warunku.
[illegible] If WYRAŻENIE Then [illegible]
WYRAŻENIE - [illegible] przyjmować wartość [illegible] jest [illegible] WYRAŻENIE jest [illegible]
[illegible] If IMIE$="Rafał" Then Print [illegible] jesteś Rafał"

If Else End If - [illegible] lecz [illegible]
[illegible] If WYRAŻENIE
Tu instrukcje [illegible]
[illegible]
[illegible]
{Else
Tu instrukcje [illegible]
[illegible]
[illegible]}
Endif
[illegible] If IMIE$="Rafał"
Print [illegible] Rafał!"
Else
Print [illegible] IMIE$ [illegible] mi nie mów."
End If

Sqr( ) - [illegible] z [illegible]
[illegible]
[illegible] Print Sqr(4)

End - [illegible]
[illegible] End

Goto - [illegible] skok do [illegible]
[illegible] Goto XX
XX - etykieta lub numer linii [illegible]
[illegible] 10

Abs( ) - [illegible] wartość absolutną [illegible]
[illegible]
[illegible] Print Abs(-10)

Gosub - Return - [illegible]
[illegible] ETYKIETA
.
.
ETYKIETA:
[illegible]
Return

Follow - instrukcja [illegible] włączenie tzw. [illegible] w którym program wykonuje się krok po kroku
[illegible] Follow {lista zmiennych}
Follow Off
[illegible] Follow ZMIENNA1 [illegible]



O PRZENOSZENIU DANYCH SŁÓW KILKA

GRAFIKA

Jakiś czas temu opisywałem technikę przenoszenia danych pomiędzy „pecetami" a Amigą – pisałem wtedy ogólnie o danych, bez rozróżnienia na grafikę czy tekst. W tej części cyklu postaram się przybliżyć sposoby, a właściwie programy do przenoszenia grafiki. Z myślą o nielicznych zapewne zainteresowanych uwzględniłem także dwa formaty rodem z Atari ST.

Prawie każdy komputer ma „swoje" formaty grafiki, przy czym Amiga, dla której istnieje jeden powszechnie respektowany standard, jest tu rzadkością. Najczęściej mamy do czynienia z wieloma formatami, z kilkoma liczącymi się i kilkunastoma odrzuconymi przez użytkowników. Dzieje się tak niestety dlatego, że jak do tej pory nie znajdował się nikt, kto - odkąd dany komputer pojawił się na rynku - wymyśliłby format na tyle praktyczny, by uniknąć tworzenia coraz to nowych z chwilą pojawienia się nowych potrzeb.

W przypadku Amigi stało się na szczęście inaczej - w chwili jej narodzin znalazło się kilku inteligentnych facetów z firmy Electronics Arts, którzy wymyślili format IFF (Interchange File Format). Jego koncepcja jest na tyle dobra, że powstrzymało to dziesiątki firm od wymyślania własnych formatów i uchroniło Amigę przed jakże uciążliwym brakiem standaryzacji. Format IFF jest bowiem tak skonstruowany, że nawet jeżeli wsadzimy do środka pliku jakieś własne, nietypowe dane (powiedzmy datę i personalia twórcy), to i tak każdy inny program odczyta plik prawidłowo. Cały dowcip polega na tym, że plik w standardzie IFF podzielony jest na segmenty - jest na przykład segment zawierający paletę, tryb pracy ekranu, dane (bitplane'y) itd. Każdy z tych segmentów ma swoje czteroliterowe oznaczenie, poprzedzające jego długość - tak więc jeżeli wstawimy do pliku segment "AUTR" i poprawnie podamy jego długość, to wszystkie programy poza naszym będą po prostu ten segment omijały i korzystały bez kłopotu z interesujących je danych. Dla przykładu, program Art Department Professional (ASDG) dodaje do pliku segment „ANNO", w którym umieszcza datę nagrania pliku i informacje o sobie.

To by było na tyle, jeśli chodzi o zalety Amigi. W przypadku „pecetów" sprawa nie jest już taka prosta. Nie ma tam bowiem żadnego dominującego standardu – formatów jest wiele, że wymienię choćby GIF, TIFF, PCX, TARGA, IFF (tak, tak). Tak więc nie wystarczy przeniesienie samego pliku (może z wyjątkiem IFF, jednak też nie zawsze) - potrzebny jest jeszcze program, który dokona konwersji. Konwersja może zresztą dotyczyć nie tylko formatu zapisu danych, stara Amiga (nie wyposażona w tzw. AA chipset, czyli nowy zestaw kości specjalizowanych) nie potrafi na przykład wyświetlić w najwyższej rozdzielczości więcej niż szesnastu kolorów i przy konwersji może zajść konieczność zmiany rozdzielczości rysunku lub liczby użytych w nim kolorów

NARZĘDZIA

Jeśli więc przebrniemy już przez etap transferu danych, staje przed nami problem, co z tymi danymi dalej zrobić - nie na wiele zda się nam plik w formacie np. TIFF, musimy przerobić go na formę zrozumiałą dla Amigi. Programy takie jak PageStream, rozpoznający niemal wszystkie liczące się formaty grafiki z innych komputerów, należą bowiem do rzadkości. W tej sytuacji wypada sięgnąć po program do konwersji i ewentualnej obróbki danych. Znalezienie potrzebnego nam oprogramowania nie powinno nastręczać większych problemów, jest go naprawdę pod dostatkiem – począwszy od do granic rozbudowanych programów w rodzaju ADPro, a skończywszy na prościutkich programikach Public Domain, takich jak HamSharp.

Art Department Professional

Program w zasadzie przeznaczony do profesjonalnej obróbki grafiki, rozpoznaje jednak kilkanaście formatów danych nie tylko graficznych, między innymi kilka formatów z IBM PC. Nie gardzi też innymi komputerami, jak Macintosh lub Sun – to już jednak temat na inny artykuł.

Tak więc przy pomocy ADPro możemy przerobić niemal dowolny format w jakikolwiek inny, w naszym przypadku IFF. To jednak, jak już wspomniałem, nie wszystko. Jeśli bowiem chcemy użyć grafiki w formacie GIF, to w przypadku, gdy jest ona w najczęściej używanej rozdzielczości 640*480*256 (640 na 480 punktów przy 256 kolorach), zachodzi konieczność zredukowania liczby użytych kolorów lub rozdzielczości. Tutaj właśnie ujawnia się największa zaleta ADPro - umożliwia on wykonanie na grafice dziesiątek operacji, w tym oczywiście skalowania i zmiany liczby użytych kolorów. Przy tym obie te operacje wykonuje on chyba najlepiej, jak to tylko możliwe. Przy skalowaniu nie ogranicza się na przykład do wycinania co n-tej linii, tylko na podstawie kilku tworzy jedną (w ten sposób na przykład nie dochodzi do sytuacji, gdy po przeskalowaniu znikają nam pojedyńcze poziome lub pionowe linie). Przy redukcji liczby użytych kolorów zachodzi natomiast konieczność zastąpienia ich innymi – także tutaj zastosowanie ADPro jest najlepszym rozwiązaniem, oferuje on bowiem aż sześć technik tzw. ditheringu, czyli mieszania różnokolorowych punktów w celu uzyskania „nowego" koloru.

ADPro ma niestety jedną wielką wadę: do wykonania najprostszej operacji potrzebuje potwornych ilości pamięci. Zresztą już po uruchomieniu, „na dzień dobry", rezerwuje sobie bufor wielkości ok. 900 KB. W praktyce jest więc niemal bezużyteczny dla posiadaczy Amig z pamięcią mniejszą niż 2 MB. Inna sprawa, że już dla samego ADPro warto, moim zdaniem, kupić duże rozszerzenie pamięci - jakość uzyskiwana przy jego pomocy jest naprawdę bezkonkurencyjna.

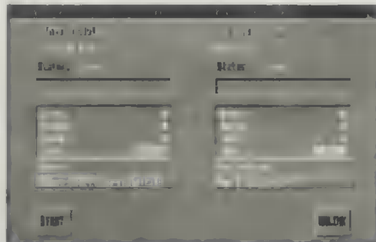
ImageLink

Program dość już leciwy (1990), służący tylko i wyłącznie do konwersji grafiki. Rozpoznaje on wszystkie najczęściej spotykane formaty, samą zaś konwersję robi nieźle, jednak jakość grafiki tworzonej przez ImageLink jest nieporównanie gorsza niż tej wygenerowanej przez ADPro. ImageLink umożliwia przeskalowanie rysunku, a także zmianę liczby użytych w nim kolorów. Skalowanie odbywa się jednak tradycyjnymi metodami (usuwana jest co n-ta linia), zaś co do koloru, to ImageLink oferuje tylko jedną technikę ditheringu (Floyd-Steinberg), uzyskując zresztą dużo gorsze efekty niż ADPro przy tym samym algorytmie. Natomiast niezaprzeczalną zaletą programu jest fakt, że zadowala się on niewielką ilością pamięci i nawet posiadacze Amigi z pamięcią 1 MB mogą go w miarę swobodnie używać. Również liczba rozpoznawanych formatów jest duża i do zwykłej konwersji (bez zmiany rozdzielczości i liczby kolorów) ImageLink nadaje się doskonale.

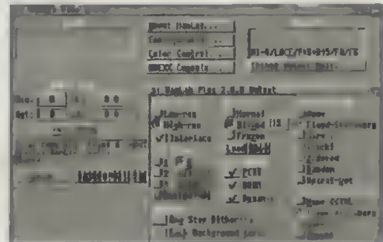
Art Department Professional



ImageLink



HamLab+





HamLab+

Pierwsza jego wersja nie należała raczej do zbyt rozbudowanych, najnowsza natomiast, HamLab Plus, jest już programem [redacted] niezłym poziomie – rozpoznaje najważniejsze formaty, umożliwia też pewną obróbkę grafiki (skalowanie, rozbudowane operacje [redacted] kolorach). Funkcją właściwą tylko dla HamLab'a jest nagrywanie grafiki [redacted] wielu „udoskonalonych" formatach IFF, jak Dynamic Hires, SHAM czy PCHG.

Wszystkie [redacted] opierają się [redacted] tej samej zasadzie: co każdą linię zmieniana jest paleta. W przypadku trybu HAM korzyść jest taka, że paleta (w HAM 4096 kolorów tworzonych jest poprzez modyfikację 16 podstawowych) dobierana jest [redacted] myślą [redacted] tylko jednej linii, [redacted] nie [redacted] całym ekranie, co umożliwia pewną poprawę jakości obrazu. Drugi tryb, Dynamic Hires, to owo słynne "4096 kolorów [redacted] najwyższej rozdzielczości". W trybie tym mamy do dyspozycji rzeczywiście 4096 kolorów, ale tylko [redacted] ramach całego ekranu - [redacted] jednej linii możemy bowiem użyć najwyżej szesnastu.

Co do jakości grafiki generowanej przez program HamLab, to niestety nie rzuca ona na kolana. Obrazek [redacted] „udoskonalonych" trybach Dynamic Hi-[redacted] lub SHAM wygenerowany przez HamLab prezentuje się gorzej niż zwykła grafika [redacted] ADPro.

Podobnie jak ImageLink, HamLab [redacted] nad ADPro pewną przewagę – zadowala się znacznie mniejszą ilością pamięci. Od programu ImageLink HamLab jest [redacted] kolei [redacted] tyle lepszy, że rozpoznaje format JPEG, charakteryzujący się niezwykłą wydajnością kompresji (nawet 20:1) przy pewnej utracie jakości.

TouchUp

Program dostarczany razem [redacted] skanerami Golden Image. Nie byłby on własciwie wart uwagi, gdyby nie fakt, że jako jedyny umożliwia konwersję dość często spotykanego [redacted] Atari ST formatu *.IMG. Ta konwersja i obsługa skanera to właściwie jedyne zastosowania, do których program ten się nadaje. Oprócz IMG TouchUp rozpoznaje także formaty PCX I TIFF, ra-[redacted] sobie jednak tylko z grafiką monochromatyczną.

Converter, HamSharp i Wasp

Te trzy programy są klasycznymi konwerterami. Wszystkie trzy napisane zostały przez amatorów I wszystkie też – może poza Wasp'em – charakteryzują się minimalnymi możliwościami. Converter i HamSharp potrafą bowiem dokonać konwersji tylko z jednego formatu. Pierwszy z nich rozpoznaje DEGAS rodem [redacted] Atari ST, drugi zaś – „pecetowy" GIF. O ile Converter radzi sobie nieźle, nie [redacted] bowiem problemów z nadmierną liczbą kolorów (Atari ST nie dysponuje, jak wiadomo, przesadnie rozbudowaną paletą), to [redacted] przypadku konwersji z formatu GIF dokonywanej przez HamSharp'a trudno zostać oczarowanym. Program automatycznie skaluje rysunek, by móc zastosować tryb HAM, jednak dithering nie prezentuje się najlepiej. Najgorzej jest [redacted] przypadku programu Wasp. Mimo że oferuje on sporo opcji, to efekty jego pracy są zdecydowanie najgorsze. Zaletami tego programu mogą być: spora szybkość działania, oszczędność [redacted] gospodarowaniu pamięcią I stosunkowo duża liczba opcji. Jeżeli jednak komuś zależy na jakości, nie tędy droga.

PODSUMOWANIE

Koniec końców, najważniejszym kryterium oceny powinna być jakość grafiki generowanej przez program. Do testu prównawczego posłużył mi 256-kolorowy rysunek w formacie GIF I [redacted] rozdzielczości 640*480 punktów. Zadaniem programów było przeskalowanie grafiki, [redacted] następnie przerobienie 256-ciu kolorów na HAM. Różnice [redacted] jakości były znaczące. Najlepiej prezentowała się twórczość ADPro. Drugie miejsce przyznałem Ham-Lab'owi, trzecie dostał ImageLink, czwarte – HamSharp. To, [redacted] wyprodukował Wasp, było wręcz ohydne.

I to wszystko w tym miesiącu. W następnym numerze (tym razem już naprawdę) znajdzie się ostatni artykuł tego cyklu, poświęcony przenoszeniu tekstów. Znajdziecie tam rady, jak przenosić teksty pomiędzy Amigą i„pecetem", jak radzić sobie [redacted] „udziwnionymi" formatami WordPerfect'a lub ChiWriter'a. Podam też kod źródłowy programu do konwersji ze standardu księdza Pikula [redacted] mazovię iodwrotnie.

ANDRZEJ BOBEK

Najważniejsze formaty rozpoznawane przez omawiane programy

PROGRAM	FORMAT						
	GIF	JPEG	TIFF	TARGA	PCX	DEGAS	IMG
ADPro	+	+	+	+	+	-	-
ImageLink	+	-	+	+	+	-	-
HamLab	+	+	+	+	+	-	-
TouchUp	-	-	+	-	+	-	+
Converter	-	-	-	-	-	+	-
Wasp	+	-	-	-	-	-	-
HamSharp	+	-	-	-	-	-	-

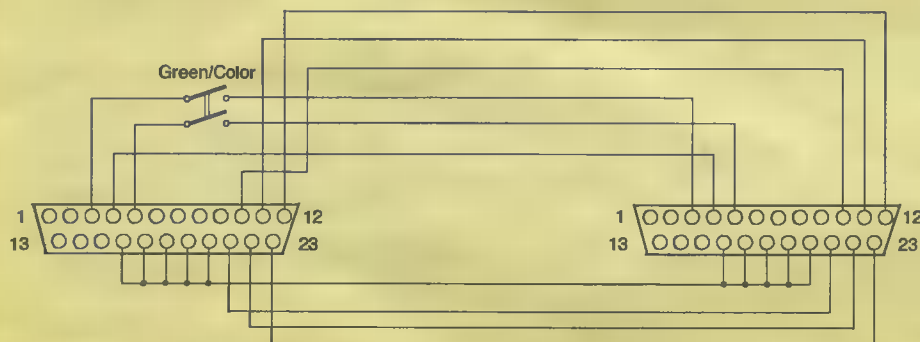
Liczba kolorów wyświetlanych p[redacted] zwykłą Amigę w poszczególnych rozdzielczościach

ROZDZIELCZOŚĆ	LICZBA KOLORÓW	PALETA
320 × 256 320 × 512	2, 4, 8, 16, 32, 64 lub 4096	4096
640 × 256 640 × 512	2, 4, [redacted] lub 16	
1280 × 256 1280 × 512	2 lub 4	

Liczba kolorów wyświetlanych przez Amigę wyposażoną [redacted] będący [redacted] ukończeniu tzw. [redacted] Chipset w poszczególnych rozdzielczościach

ROZDZIELCZOSC	LICZBA KOLORÓW	PALETA
320 × 256 320 × 512	2, 4, 8, 16, 32, 64, 128, 256, 4096 lub 256.000	16.777.216
640 × 256 640 × 512		
1280 × 256 1280 × 512		

PRZEŁĄCZNIK „GREEN" dla AMIGI



Niektóre monitory (np. nowsze wersje monitora Commodore 1084S) są wyposażone [redacted] przełącznik „Green" umożliwiający przejście do trybu monochromatycznego [redacted] odcieniu zielonym. Jest to przydatne, zwłaszcza gdy korzystamy np. [redacted] edytorów tekstu – przy dłuższej pracy nie męczy się wzrok. Właśnie przez wzgląd na oczy chciałbym zaproponować Czytelnikom całkowicie samodzielne wykonanie takiego przełącznika.

Aby nie naruszać konstrukcji monitora czy telewizora, przełącznik wykonuje się jako interfejs włączany pomiędzy Amigę [redacted] kabel monitora lub modulator TV. Do wykonania przełącznika będą nam potrzebne: złącze 23-stykowe typu Canon, męskie, takie samo złącze żeńskie, przełącznik dwubiegunowy (np. Isostat) oraz lutownica i trochę przewodu jednożyłowego.

Przełącznik ma bardzo prostą konstrukcję, co jest okupione podstawową wadą: kolory nie zawierające [redacted] swoim składzie barwy G, czyli zielonej, nie będą widoczne [redacted] ekranie. Co ciekawe, taki typ przełącznika stosuje firma Commodore – do pracy [redacted] tekstami [redacted] zupełności to wystarcza. Jest zrozumiałe, że do prac graficznych należy przełączyć monitor [redacted] tryb „Color". Oczywiście można bez większych problemów wykonać przełącznik pozbawiony wymienionej wady, lecz jest to już temat [redacted] oddzielny artykuł.

JERZY DUDEK

LhA

KAŻDY TO POWINIEN MIEĆ

LhA jest jednym z najbardziej popularnych archiwizerów w świecie Amigi, a także w świecie komputerów kompatybilnych (zgodnych) z IBM PC. Szczególnie przydatny jest dla osób mających modem, a także dla wszystkich cierpiących na ciągły brak miejsca na dyskietkach (jak niżej podpisany). Dzięki niemu możemy zaoszczędzić nawet i 50% miejsca! Pozwala przechowywać w formie archiwum programy aktualnie nie używane, lecz mogące się przydać w przyszłości. Program ten jest zgodny z programami LhArc, LhArcA i LZ na Amigę, a także z programami LhArc i LhA dla komputerów IBM-PC.

LhA uruchamia się na każdym komputerze Amiga wyposażonym co najmniej w KickStart 1.2, 512 KB pamięci RAM i jedną stację dysków. Pracując pod kontrolą KickStartu 1.2 lub 1.3 używa biblioteki „arp.library", która powinna się znaleźć w katalogu „LIBS:" dyskietki systemowej (tej, z której uruchamiamy system), bądź dysku twardego. LhA jest programem, którego można używać jedynie z CLI (bądź Shell), co oznacza, że niemożliwe jest jego uruchomienie z Workbench'a (za pośrednictwem ikony).

Składnia wywołania programu wygląda następująco:
LhA [-opcje] (komenda) (archiwum) [katalog] [pliki]
gdzie nawias okrągły oznacza konieczność podania polecenia, a nawias kwadratowy, że polecenie nie jest konieczne do poprawnego działania programu (zależy to od wykonywanej komendy).

Chciałbym teraz omówić pokrótce kilka najważniejszych i najczęściej używanych poleceń programu LhA. Pełną listę komend i opcji można uzyskać pisząc w oknie CLI samą nazwę programu i wciskając RETURN. Każdą wykonywaną funkcję programu można w każdej chwili zatrzymać wciskając na raz klawisze CTRL i C.

KOMENDY

a – dodawanie plików do archiwum, w przypadku nie znalezienia archiwum program tworzy nowe o podanej nazwie.
m – komenda mająca podobne zastosowanie co komenda „a", lecz pliki dodawane do archiwum zostaną automatycznie skasowane ze źródła (skąd zostały dodawane).
d – kasowanie plików z archiwum.
e – ekstrakcja („wyciąganie") plików z archiwum.
x – komenda podobna do „e", z tym że pliki zostaną rozpakowane wraz ze ścieżką dostepu do nich (to samo co komenda „e" wraz z opcją „-x").
l – przeglądanie archiwum, komenda ta pokazuje nam listę plików w archiwum.
v – opcja podobna do „l", lecz wyświetlone zostaną pliki wraz z pełną ścieżką dostępu do każdego z nich.
t – testowanie archiwum – komenda ta powoduje, że program przetestuje dane archiwum i sprawdzi poprawność zapisu zawartych w nim danych.

OPCJE

Przy podawaniu opcji należy pamietać o poprzedzeniu ich znakiem „-" (minus), a także o tym, że program rozróżnia duże i małe litery, oraz że niektóre opcje wymagają dodatkowych danych. Możliwe jest podawanie kilku opcji przy jednym wywołaniu LhA, lecz należy pamiętać, aby odzielić je od siebie przynajmniej jedną spacją. W nawiasie kwadratowym podane jest, przy jakiej komendzie można używać danej opcji.

-D(num) [wszystkie]
Powoduje zmianę pokazywania postępów w pracy programu (tzw. progress indicator), dostepne są następujące jego rodzaje:
0 – ustawiony domyślnie, (xxxxxx/yyyyyy) gdzie:
xxxxxx = ilość bajtów spakowanych (rozpakowanych),
yyyyyy = długość niespakowanego pliku,
1 – pokazuje jedynie obracającą się kreskę (-),
2 – wskaźnik procentowy,
3 – wskaźnik pasmowy (liniowy).
Przykładowe użycie:
LhA -D3 a RAM:MyArch SYS:C/

-e [a,m]
Program archiwizuje także puste katalogi, domyślnie opcja ta jest wyłączona (program nie archiwizuje pustych katalogów).

-f [wszystkie]
Program ignoruje komentarze do plików.

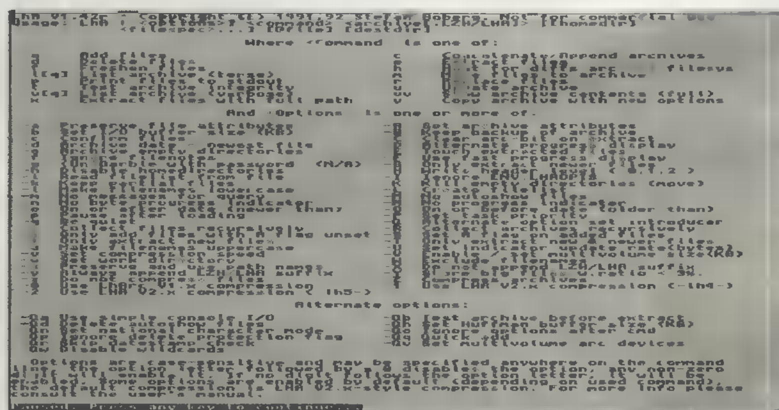
-l [wszystkie]
Program zmienia wszystkie duże litery w nazwach plików na małe.

-L(path) [wszystkie]
Program stworzy listę plików, na których przeprowadzał operację, przykładowe użycie:
LhA -Lram:Lista a ram:Archiwum SYS:L/
Po takim wykonaniu LhA w „RAM:" znajdować się będzie plik o nazwie „Lista" o przykładowej strukturze:
L/Port-Handler
L/DiskValidator
L/Ram-Handler

-M [e,x]
Program przy rozpakowywaniu archiwum nie będzie pokazywać plików mających rozszerzenie ".displayme".

-N [wszystkie]
Program nie będzie podawać żadnych informacji o postępach w pracy.

-p [wszystkie]
Program po załadowaniu się z dyskietki zaczeka na wciśnięcie dowolnego klawisza przez użytkownika – jest to szczególnie przydatna opcja posiadaczy jedynie jednej stacji dysków. Mogą oni teraz zmienić dyskietkę i dopiero wtedy pozwolić pracować LhA.



-P(num) [wszystkie]
Ustawia priorytet, z jakim będzie pracować LhA, w zakresie od -5 do 5 (liczby całkowite), domyślnie ustawiona jest 0. Od priorytetu zależy, ile czasu poświęci mikroprocesor na pracę LhA, spowalniając lub przyśpieszając inne zadania, np.:
LhA -P-3 a RAM:Archiwum SYS:L/
spowoduje uruchomienie LhA z piorytetem równym -3.

-q [wszystkie]
Opcja ta spowoduje, że LhA nie będzie wysyłać żadnego komunikatu do okna CLI.

-r [a,m]
Doda do archiwum nie tylko pliki znajdujące się we wskazanym katalogu, ale także pliki ze wszystkich podkatalogów znajdujących się w tym katalogu, np.:
LhA -r a RAM:MyArch DF1:
spowoduje archiwizację wszystkich plików znajdujących się na dyskietce w stacji oznaczonej jako „DF1:" (także tych w podkatalogach).

-R [wszystkie]
Opcja podobna do „-r", tyle że działa jedynie na archiwa, np.:
LhA -R l DH0:pliki/f*
pokaże zawartość wszystkich archiwów zaczynających się na literę f w katalogu „DH0:pliki" i podkatalogach.

-u [wszystkie]
Opcja podobna do „-l", jednak o przeciwstawnym działaniu.

-V(num) [a,m]
Opcja ta spowoduje, że w przypadku zapełnienia się miejsca na danej dyskietce LhA poprosi nas o jej zmianę na następną. Można podać ilość miejsca do zapełnienia (w kilobajtach), bądź podać „a" w celu zapełnienia całego dostępnego wolnego miejsca np.:
LhA -r -Va a DF1:Arch DH0:Tools/
spowoduje, że w przypadku zapełnienia się całego wolnego miejsca na dyskietce w napędzie „DF1:", LhA poprosi nas o jej zmianę na następną. Należy pamiętać, że z takiego archiwum wolno później jedynie rozpakowywać pliki lub je wylistować.

-x [wszystkie]
Program zapamiętuje ścieżkę dostępu do każdego pakowanego pliku, bądź jej używa w przypadku rozpakowywania archiwum i tworzy odpowiednie katalogi. Opcja ta jest automatycznie włączana w przypadku używania opcji „-r".

-X [wszystkie]
Nie jest dodawane rozszerzenie „.lzh" bądź „.lha" do podanej nazwy archiwum.

-y [wszystkie]
Zawsze dodawane jest rozszerzenie „.lzh" bądź „.lha" do podanej nazwy archiwum, nawet jeśli ono takowe ma.

-Y [a,m]
Pliki, w których stosunek długości po spakowaniu do długości przed spakowaniem jest mniejszy od 3%, nie będą

dodane do archiwum jako spakowane, co zlikwiduje czas potrzebny na ich rozpakowanie (szczególnie przydatne przy wolnych Amigach).

-z [a,m]
Przy dodawaniu do archiwum, pliki nie będą pakowane.

-0 [a,m]
Używany będzie algorytm kompresji tzw. „-lh1-", najmniej efektywny, lecz zgodny z archiwami programu LhArc, dodawane będzie rozszerzenie „.lzh".

-1 [a,m]
Używany będzie algorytm kompresji tzw. „-lh4", minimalnie szybszy przy pakowaniu niż „-lh5-", lecz dający gorsze wyniki i wolniejszy przy rozpakowywaniu, dodawane będzie rozszerzenie „.lha".

-2 [a,m]
Używany będzie algorytm kompresji tzw. „-lh5-", minimalnie wolniejszy przy pakowaniu niż „-lh1-", lecz dający najkrótsze archiwa i najszybszy przy rozpakowywaniu, dodawane będzie rozszerzenie „.lha". Ten algorytm kompresji jest ustawiony domyślnie.

Dla LhA powstały również nakładki graficzne, ułatwiające pracę z tym programem. Przykładem jednej z nich może być program LhAX (ok. 7 KB długości). Umożliwia on rozpakowanie lub przetestowanie dowolnego archiwum, wyświetlenie jego zawartości – cały ten program używa biblioteki „asl.library" (OS2.0).

Na koniec należy dodać, że program LhA rozprowadzany jest jako „Shareware", co wymaga od użytkownika zapłacenia autorowi programu opłaty za jego używanie (ok. $20), a to upoważnia do otrzymywania najnowszych wersji LhA, a także do pomocy autora w przypadku złej pracy programu. Jako ciekawostkę mogę podać, że autor programu – Stefan Boberg – ma 19 lat.

HIGHTOWER

Słowniczek

ARCHIWUM – plik zawierający kilka plików (najczęściej w formie spakowanej) wraz z informacją o nich (nazwa, komentarz, data).
ARCHIWIZACJA – proces tworzenia archiwum.
ARCHIWIZER – program obsługujący archiwa (tworzy je, listuje, itd.).
ŚCIEŻKA DOSTĘPU (ang. path) – każdy plik ma swoją ścieżkę dostępu pokazującą, gdzie się znajduje. Np. dla pliku o nazwie „Fish.lha" znajdującego się na dyskietce w stacji o symbolu „DF1:" w katalogu „Archiwa" i podkatalogu "Amiga", ścieżka dostępu ma postać:
DF1:Archiwa/Amiga/Fish lha



BEZKARNOŚĆ ?

Zdarzyło się to na giełdzie komputerowej przy ul. Grzybowskiej w Warszawie. Przybyli na nią wtedy zarówno ci, którzy chcieli się pozbyć towaru, jak i ci, którzy chcieli się pozbyć pieniędzy na rzecz tych pierwszych. Wymienionym towarem obok wszelkiego sprzętu było także oprogramowanie, w sporej mierze pirackie. Prawo polskie jeszcze nie uważa oprogramowania za rzecz chronioną, tak więc sprzedawcy nielicencjonowanych programów zarabiają dziennie wcale niezłe pieniądze.

W pamiętnym dniu 12 września br. sielankowy wręcz nastrój na giełdzie zakłóciło przybycie przedstawicieli firmy IPS, prowadzącej dystrybucję licencjonowanego oprogramowania, oraz przedstawicieli Straży Miejskiej. Wśród piratów zapanował popłoch. Niektórzy natychmiast spakowali się i wycofali cichcem, inni, zasłoniwszy dyskietki i komputer (działający jako kopiarka), wywiesili ogłoszenia typu: MONITOR SPRZEDAM W KAŻDĄ SOBOTĘ I NIEDZIELĘ.

Okazało się, że Straż Miejska wymagała od handlujących okazania zezwoleń na prowadzenie działalności gospodarczej, natomiast panowie z IPS-u przedstawili listę z tytułami zastrzeżonych programów. Kiedy przedstawiciele władzy i IPS-u opuścili targowisko, wśród sprzedawców rozgorzała ostra polemika na temat bezprawia, ochrony oprogramowania i ogólnie na temat „akcji pacyfikacyjnej", jakiej śmieli się dopuścić panowie z warszawskiej firmy.

Udałem się zatem do jej szefa – Grzegorza Onichimowskiego, w celu uzyskania bliższej informacji na temat owego wydarzenia.

PLiS: Skąd wziął się pomysł takiej „akcji"?

GO: Firmy, które reprezentujemy, żądają od nas zahamowania masowej sprzedaży nielegalnych kopii ich produktów. Z uwagi na brak prawnej ochrony oprogramowania jest to bardzo trudne. Na szczęście znaleźliśmy zrozumienie wśród warszawskich władz (Służba Miejska) i wraz z nimi udaliśmy się na giełdę przy ulicy Grzybowskiej. Nasze kroki skierowaliśmy ku sprzedawcom oferującym gry głównie na Amigę i komputery PC. Na razie upominaliśmy piratów, o konsekwencjach grożących z tytułu przywłaszczania sobie cudzej pracy. Natomiast oni po otrzymaniu od nas listy z kilkudziesięcioma tytułami gier obiecali wycofać je ze sprzedaży.

PLiS: Konfrontacja z handlarzami przebiegła zadziwiająco spokojnie, zatem jaką funkcję pełnili przedstawiciele władzy? Doszły nas słuchy, że kilka osób zostało „spisanych".

GO: Panowie ze straży miejskiej spisywali personalia sprzedawców, którzy nie posiadali zezwolenia na prowadzenie działalności gospodarczej. Więc ci, którzy sądzili, że płacąc placowe zyskują takie zezwolenie, srodze się zawiedli.

PLiS: To co zaszło w sobotę na giełdzie wywołało jednak falę protestów i wzburzenia. Zarzucano firmie IPS bezprawne, w świetle obowiązującego prawa, egzekwowanie praw autorskich na gry. Piraci twierdzili, że mają na razie takie samo prawo do sprzedaży tych gier jak i firma IPS, więc nie powinno się od nich wymagać zaprzestania handlu. Ponadto wiadomo, że program oferowany przez pirata kosztuje bardzo niewiele w stosunku do jego licencjonowanej wersji, co nie jest obojętne dla potencjalnego klienta, który coraz to dokładniej waży zawartość swojej kieszeni. Co Pan o tym sądzi?

GO: Program oferowany przez pirata jest rzeczywiście tańszy od oferowanego przez nas. Dzieje się tak dlatego, że towar, czyli w tym przypadku program, sprzedawany na giełdzie jest jedynie fragmentem tego, co sprzedaje IPS. Mówiąc to mam na myśli dokładną instrukcję obsługi, w 100% działający program, możliwość zakupienia nowszych wersji po niższych cenach. Tego wszystkiego pirat nie jest w stanie zapewnić.
Druga sprawa to podatki. Człowiek sprzedający cudze programy na giełdzie najczęściej nie płaci podatków i w ten sposób wygrywa z nami konkurencję. Zawsze powtarzam, że jesteśmy w stanie konkurować z rynkiem pirackim, pod warunkiem, że będą nim rządziły te same prawa podatkowe.
Wreszcie trzecia sprawa to prawa autorskie. Sprzedawcom kradzionego oprogramowania wydaje się, że skoro nie ma u nas jeszcze znowelizowanej ustawy o prawach autorskich, to nic im nie grozi. Otóż mylą się. Obok programu jako rzeczy w zasadzie nienamacalnej istnieją znaki firmowe, zastrzeżone nazwy itp., które jak najbardziej podlegają ochronie. Właśnie na tej podstawie dochodzić można swoich praw. Oczywiście można usunąć te elementy z programu, co nie daje gwarancji jego pełnego działania, a pożera często bardzo dużo czasu i pracy. Takie rozwiązanie jest zatem nieopłacalne.

PLiS: Pozostaje jeszcze sprawa przyzwyczajenia. Od momentu wejścia na nasz rynek komputerów obserwuje się bardzo ciekawe zjawisko: ktoś kupuje komputer i natychmiast zdobywa do niego niesamowite ilości programów. Zapytany o prawa autorskie, prawo własności itp. odpowiada, że skoro

wydał kupę pieniędzy na sprzęt, to programy mu się należą.

GO: To już jest sprawa tzw. moralności. Czy jeżeli zazdrości się komuś mercedesa, a nie stać na jego kupno, to czy trzeba od razu wybijać szybę łomem i kraść samochód? Do takiego samego poziomu można by sprowadzić sprawę kradzionego oprogramowania. A warto nadmienić, że złodziejem, mimo, że często nieświadomie, zostaje każdy, kto takiego oprogramowania używa.

PLis: W związku z tym, jeżeli chce się używać komputera „legalnie", to trzeba zakupić licencjonowane oprogramowanie. Jednakże w obecnej chwili na naszym rynku albo go nie ma (ew. sieroce), albo kosztuje niesamowite, jak na kieszeń przeciętnego Polaka, pieniądze. Czy wejście ustawy w takiej sytuacji ma sens?

GO: Oczywiście, że tak. Przecież pod pojęciem programów nie można rozumieć tylko gier tworzonych za granicą. W Polsce także powstają doskonałe programy np. na komputery Amiga i PC. Proszę sobie wyobrazić, że napisanie porządnej gry trwa od kilku miesięcy do kilku lat(!). A każdy przecież chce za swoją ciężką pracę dostać wynagrodzenie. Tymczasem dzięki działalności rynku pirackiego większość zysków zgarniają właśnie piraci, a nie prawowity twórca.
Druga sprawa to cena oprogramowania. Fakt, że nie jest ona najniższa w stosunku do zarobków „szarego człowieka". Zważmy jednak, że są to ceny często wielokrotnie niższe od zachodnich. Poza tym, za granicą za dobry program płaci się średnio 30% ceny komputera, podczas gdy w Polsce ok. 10%. Należy wyjść z założenia, że skoro kogoś było stać na mercedesa, to będzie go też stać na kupno benzyny do niego.

PLis: Mówimy tutaj głównie o grach na „mercedesy" wśród komputerów czyli PC, Amigę i Atari ST. A co z Commodore 64? Oferty polskich firm nie rozpieszczają zbytnio posiadaczy tych modeli komputerów. Czy piraci sprzedający programy do „ośmiobitowców" też mają się czuć „zagrożeni"?

GO: Z oprogramowaniem do tych komputerów rzeczywiście jest niewesoło. Dzieje się tak dlatego, że sprowadzanie gier np. na C-64 jest po prostu nieopłacalne. Taki program kosztowałby dość dużo w stosunku do ceny komputera. Po za tym coraz to więcej posiadaczy komputerów oraz chcących je kupić kieruje się w stronę 16-bitowców, a co za tym idzie – większych możliwości.
PLis: Czy należy zatem rozumieć, że sprzedawcy programów na C-64 mogą się czuć bezkarni?

GO: Tego bym raczej nie chciał powiedzieć. Z chwilą wejścia nowej ustawy (być może jeszcze w tym roku – dop. aut.) o prawach autorskich sprzedawanie pirackich programów na jakiekolwiek komputery będzie niezgodne z prawem.

PLis: Co więc mają zrobić ci, którzy chcą sobie pograć, lecz nie będą mieli pieniędzy na zakup gier?

GO: Jest wiele legalnych sposobów zarobienia pieniędzy lub zdobycia programów. My na przykład bardzo chętnie widzielibyśmy młodych ludzi zainteresowanych zarobieniem na własną grę. Mogliby oni wziąć od nas kilka programów, sprzedać je np. kolegom w szkole, za co otrzymaliby upatrzoną dla siebie grę. Jeżeli okazałoby się obrotni i utworzyli sieci sprzedaży tych gier, to skłonni jesteśmy nawet iść na jeszcze większe ustępstwa.

PLis: Co na zakończenie przekazałby pan naszym Czytelnikom?

GO: Cokolwiek robimy, mam na myśli giełdę, nie robimy tego dla, że tak powiem, moralizatorskich celów lecz czysto komercyjnych, ale warto by było, żeby owa przypowieść o mercedesie zapadła nam w pamięć. Musimy sobie zdawać sprawę, że cokolwiek robimy i czy to jest zupełnie materialne lub tylko częściowo, tak jak np. piosenka, wiersz czy właśnie program, ma dla nas jakąś wartość i ktoś tę wartość musi uszanować niezależnie od wszystkich kwestii prawnych. Tak właśnie jest w wyżej rozwiniętych krajach. My będziemy swoich praw konsekwentnie dochodzić. Nie możemy się bowiem zgodzić na to, aby tę ciężką przecież pracę ludzi, obojętnie czy za granicą czy tutaj w Polsce, oddawać za darmo. To jest istotne, gdyż większość programistów po prostu nie ma innego źródła dochodów. Będziemy starali się stworzyć możliwości, aby jak najwięcej osób mogło na te swoje już prawdziwe, legalne programy zarobić, właśnie poprzez tworzenie sieci, dogadywanie się ze szkołami i klubami komputerowymi.

PLis: Dziękuję za rozmowę.

rozmawiał
PIOTR LISZEWSKI

Cześć! Spotykamy się już po raz trzeci (i jak się okazuje nie ostatni). W poprzednim odcinku kolega o szwarcenegerowatym przezwisku objaśnił Wam zasadę wczytywania i zapisywania programów na dyskietkach i kasetach magnetofonowych. Dziś jeszcze raz powrócimy do obsługi nośników informacji. Oprócz tego omówię funkcje portów wejścia/wyjścia, budowę C-64 i wyjaśnię, co to jest interpreter.

Poznaliście już zasadę wczytywania programu z dyskietki przy pomocy rozkazu LOAD „nazwa",8 <RETURN>. Czasami jednak stosuje się ten rozkaz w postaci

LOAD „nazwa",8,1 <RETURN>

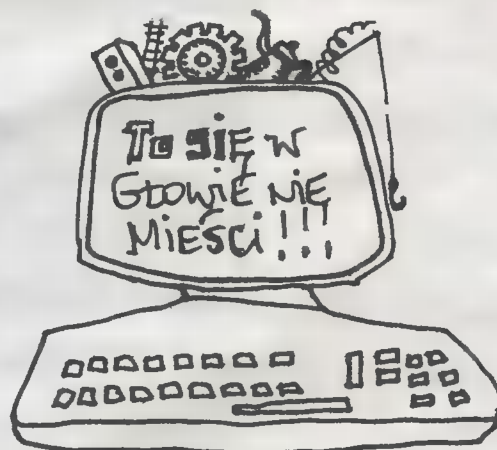
I tu nasuwa się pytanie: co robi ta jedynka na końcu? Otóż dzięki niej program wczytuje się w dokładnie określony obszar pamięci – o ile oczywiście był zapisany poleceniem SAVE „nazwa",8,1 (a więc również z jedynką na końcu). Jedynka pełni rolę znacznika i informuje komputer, że wczytywanie ma się odbywać inaczej niż zwykle.
Na razie zostawimy ten temat. Poruszyłem go tylko po to, abyście nie mieli żadnych kłopotów przy wczytywaniu programów – teraz będziecie już wiedzieć co zrobić, gdy program nie będzie chciał wczytać się po wykonaniu LOAD „nazwa",8.
Teraz powiemy sobie troszkę o zapisie w tzw. TURBO. Tak nazywany jest program, który pozwala na wczytanie i zapisanie programu na kasecie z ok. 10-krotnie większą prędkością. Ponieważ większość programów na C-64 nagrywana jest w TURBO, warto się nauczyć jego obsługi. Program ten umieszczany jest obecnie w prawie każdym module (ang. cartridge), lecz nie obsługuje się go jednakowo (ze względu na różne wersje). Przykładowo w modułach FINAL II, III i ACTION REPLAY wczytanie programu z magnetofonu odbywa się po podaniu rozkazu w postaci:

LOAD „nazwa",7 <RETURN>

natomiast w kartach X, BLACK BOX, POWER, MAGIC program wczytujemy sekwencją:

←L

Zapis danych w obu przypadkach odbywa się analogicznie, jedynie zamiast polecenia LOAD piszemy SAVE lub samą literę S.
Oczywiście istnieją też programy przyspieszające odczyt/zapis danych na stacji dysków. Niektóre z nich pozwalają uzyskać nawet 20-krotne przyspieszenie (moduł ACTION REPLAY, programy FAST HACK'EM, FILE COPIER itd.). Programy tego typu określa się mianem „dopalaczy".
Czy zastanawialiście się dlaczego komputer rozumie polecenia wydawane przez Was? Ot, na przykład takie wczytanie programu z magnetofonu. Wystarczy napisać LOAD i komputer od razu wie, że musi ładować program. Jeżeli sądzicie, że komputer rozumie po angielsku, to się bardzo mylicie. Dla komputera zrozumiała jest dopiero informacja



rys. D. DUCKI



KĄCIK POCZĄTKUJĄCEGO (3)

zapisana w kodzie maszynowym reprezentowanym przez rzędy zer i jedynek. Taki zapis programu czy polecenia byłby jednak zbyt trudny i niezrozumiały dla każdego (nie tylko początkującego) użytkownika C-64 i dlatego właśnie wymyślono interpreter języka BASIC. Pełni on rolę tłumacza na linii użytkownik - procesor i zamienia wpisane przez człowieka instrukcje typu LOAD, SAVE (a więc po prostu słowa języka angielskiego) na odpowiednie, następujące po sobie sekwencje zer i jedynek.

INTERPRETER

Jest tu specjalny program ułatwiający porozumiewanie się użytkownika z komputerem. Głównym zadaniem interpretera jest tłumaczenie poleceń zgodnych z językiem BASIC (np. RUN, PRINT) na postać zrozumiałą dla procesora (zera i jedynki). Interpreter pełni oprócz tego kilka innych funkcji, m.in. kontroluje poprawność wydanych instrukcji. W przypadku, gdy są one nieprawidłowe, interpreter wyświetla odpowiednie komunikaty, naturalnie wg konwencji BASIC'owej - po angielsku.

Warto by teraz choć z grubsza zorientować się, jak zbudowany jest C-64. Najważniejszy układ to mikroprocesor MOS 6510 (w starszych wersjach C-64 MOS 6502). Ta mała „kostka" jest sercem komputera: zarządza, liczy i rozdziela zadania dla innych układów. Potrafi w ciągu jednej sekundy wykonać ponad 100 tysięcy podstawowych operacji!

Sam mikroprocesor jest niczym bez układów wejścia/wyjścia (ang. I/O - Input/Output). W C-64 są to dwa układy scalone CIA 6526 (Complex Interface Adapter). Odpowiadają one m.in. za poprawne odczytywanie danych z klawiatury, joysticków, ze stacji dysków i magnetofonu.

Kolejne dwa bardzo ważne układy to VIC II 6567 (Video Interface Controller) i SID 6582 (Sound Interface Device). VIC II odpowiada za obraz generowany przez C-64, a więc za grafikę i kolory. Ponadto umożliwia równoczesne wyświetlanie maksymalnie ośmiu duszków (ang. sprites), czyli ruchomych obiektów. Dzięki układowi VIC II C-64 dysponuje niebagatelną rozdzielczością obrazu: 320*200 punktów.

Natomiast układ SID decyduje o możliwościach dźwiękowych C-64, a są one całkiem niezłe: ośmiooktawowa skala, można odtwarzać trzy głosy jednocześnie, cztery rodzaje obwiedni do wyboru, możliwość synchronizacji głosów i dowolnej zmiany barwy brzmień.

Pozostała ostatnia z zapowiedzianych na dziś spraw: porty. C-64 ma ich aż osiem, omówię je pokrótce.

PORT

Portem w terminologii komputerowej nazywa się gniazdo połączeniowe, poprzez które komputer może komunikować się (wymieniać dane) z urządzeniami zewnętrznymi.

1. MEMORY EXPANSION PORT – port rozszerzenia. W to gniazdo wkładamy głównie moduły oraz rozszerzenia pamięci.

2. RF, TV PORT – gniazdo modulatora umożliwiające podłączenie telewizora do C-64 kablem antenowym.

3. VIDEO – port, do którego podłączamy monitor lub wzmacniacz. Tędy przesyłane są sygnały wizyjne i foniczne niskiej częstotliwości, możliwe jest więc podłączenie komputera do telewizora wyposażonego w złącze Audio/Video.

4. SERIAL – tzw. port szeregowy. Do tego gniazda podłączamy m.in. stację dysków i drukarkę. Transmisja danych odbywa się tu szeregowo, tzn. dane przesyłane są do komputera i wychodzą z niego kolejno, po jednej linii („gęsiego").

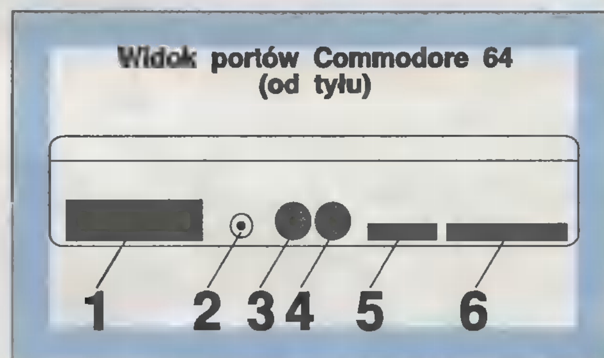
5. CASSETTE PORT – złącze krawędziowe pozwalające na podłączenie oryginalnego magnetofonu (np. DATACORDER, DATASETTE).

6. USER PORT – port użytkownika. Do tego złącza krawędziowego można podłączyć wiele urządzeń, np. drukarkę przystosowaną do pracy w standardzie CENTRONICS (równoległa transmisja danych), modem, programator pamięci EPROM. Poprzez USER PORT można też sterować urządzeniami elektronicznymi, np. oświetleniem choinkowym.

7 i 8 – gniazda joysticków – najważniejsza (oprócz samego komputera oczywiście) rzecz dla graczy. Można tu również podłączyć wiosełka (ang. paddles), wykorzystywane w niektórych grach, oraz mysz (niezbędną w przypadku korzystania z programu GEOS).

I to by było na tyle w tym odcinku. W kolejnej części będzie kilogram (lub dwa) informacji na temat urządzeń zewnętrznych, czyli co jest lepsze i dlaczego. Na zrazie!

PIOTR LISZEWSKI



NAJPROSTSZA BAZA DANYCH DLA C-64

W lipcowym numerze C&A prezentowany był artykuł pod tym samym tytułem dotyczący komputera C-128. Wszyscy, którzy biegle znają zawiłości BASIC'a V2.0 i V7.0, mogli sobie sami przerobić zamieszczone programy w celu przystosowania ich do C-64. Jednakże liczne telefony i listy docierające do redakcji z pytaniem „A co z bazą danych dla C-64?" skłoniły mnie do samodzielnego przetłumaczenia programów oraz napisania niniejszego artykułu.

Tak więc czas na stworzenie na dyskietce zbioru typu REL o nazwie „ADRESY" (nasza baza danych dla C-64 będzie również pełniła rolę książki adresowej). Nie muszę chyba przypominać, że zbiór REL jest stworzony właśnie na potrzeby baz danych, że zapis w takim zbiorze nosi nazwę rekordu i ma przyporządkowany oddzielny numer. Zaczynamy.

W C-128, aby stworzyć zbiór typu REL, wystarczyło otworzyć tylko jeden kanał transmisji ze stacją dysków. W C-64 należy otworzyć dodatkowy kanał, za pomocą którego będziemy wysyłali i odbierali komunikaty ze stacji dysków (patrz linia 210). W linii 220 otwieramy zbiór REL o nazwie ADRESY. Instrukcja CHR$(80) informuje komputer (a zarazem stację dysków), że każdy rekord będzie zawierał 80 znaków. Maksymalna długość rekordu, podobnie jak w C-128 wynosi 254 znaki, a minimalna - 2 znaki. Chyba każdy pamięta, iż ustalenia o długości rekordu dokonujemy tylko raz (w trakcie tworzenia zbioru) oraz, że po nazwie zbioru przecinek, literka „l" oraz przecinek są niezbędne (podczas pisania baz danych początkujący programiści bardzo często o tym zapominają).

W linii 230 poprzez pomocniczy kanał komunikacji ze stacją dysków informujemy, który rekord będziemy odczytywać lub zapisywać. Składnia tej instrukcji:

PRINT #15,„P"+CHR$(96+nr kanału)+CHR$(młodszy bajt rekordu)+CHR$(starszy bajt rekordu)+CHR$(nr znaku)

Numer kanału określa kanał, przez który komunikujemy się ze stacją dysków - w naszym przypadku 2. Numer znaku określa, od którego znaku komputer ma odczytać bądź zapisać dane w rekordzie. W naszym przypadku jest to 1, gdyż będziemy zapisywać dane od pierwszego znaku.

A teraz nie obyci z C-64 krzykną: „Lecz co to jest młodszy i starszy bajt rekordu?" Jak wiadomo komputer w jednej komórce pamięci może zapisać liczbę z zakresu 0-255, a nasza baza danych może mieć więcej rekordów niż 255 (maksymalnie 255*255 tj. 65025). Tak więc komputer, aby zapisać liczby z zakresu 256 do 65025 potrzebuje 2 bajtów pamięci. No, ale dosyć tych teoretycznych rozważań jak przeliczyć daną liczbę na młodszy i starszy bajt. Wystarczy skorzystać z poniższego wzoru:

SB=INT(WARTOŚĆ/256):MB=WARTOŚĆ-(SB*256)

Gdzie:

- WARTOŚĆ – wartość przeliczana na starszy i młodszy bajt
- MB - młodszy bajt
- SB - starszy bajt

UWAGA:

W omawianych tu programach przyjęto, że ich użytkownik nie będzie zakładał baz danych zawierających więcej niż 255 rekordów, co w większości przypadków jest wystarczające. Jeżeli jednak ktoś chciałby zakładać większe bazy danych, powinien skorzystać z powyższego wzoru.

Wróćmy jednak do naszego programu. W linii 240 następuje zapis w rekordzie nazwiska i adresu w postaci wcześniej zdefiniowanych zmiennych. Kod CHR$(13) (zmienna P$) jest równoznaczny z naciśnięciem klawisza RETURN (a więc dane będą zapisywane od nowego wiersza). Kod ten traktowany jest także jako znak, zatem zamiast 80 znaków w rekordzie będziemy mieć tylko 78 efektywnych.

W linii 250 zamknięty zostaje kanał łączności ze stacją dysków, nasz zbiór typu REL został utworzony. Pozostaje tylko zapisać w pierwszym rekordzie, jaką ma długość. W tym celu w linii 270 ponownie otwieramy nasz zbiór „ADRESY", w linii 280 informujemy stację dysków, że zapisywać dane będziemy w rekordzie 1. W linii 290 zapisujemy aktualną liczbę rekordów (2). W linii 300 możemy zamknąć naszą bazę danych oraz kanał pomocniczy.

Uff... W końcu mamy założoną bazę danych, lecz pora dopisać do niej parę rekordów. Aby to zrobić, wystarczy uruchomić program z listingu 2.

Po uruchomieniu wystarczy podawać nowe nazwiska i adresy – liczba rekordów będzie uzupełniana automatycznie. Mam nadzieję, że analiza tego programu nie nastręczy trudności.

Program z listingu 3 wyszukuje podane przez nas nazwisko i po znalezieniu go wyświetla cały rekord. Można także podać tylko pierwszą literę nazwiska, komputer wyświetli wtedy wszystkie nazwiska występujące w naszej bazie danych zaczynające się na tę literę.

Komentarza wymagają linie 300 i 310, w których następuje sprawdzenie, czy wszystkie rekordy zostały przeszukane w celu odnalezienia danego nazwiska. W linii 300 poprzez kanał pomocniczy pobierane są komunikaty zwrotne stacji dysków. Linia 310 sprawdza, czy przypadkiem nie wystąpił błąd o numerze 50, tj. RECORD NOT PRESENT (brak rekordu). Jeżeli tak, to baza danych zostaje zamknięta i wyświetlany jest komunikat „NAZWISKO NIE ODNALEZIONE".

Program z listingu 4 wyświetla na ekranie dany rekord.

Linie 280 i 290 mają na zadanie sprawdzić, czy dany rekord istnieje.

Pozostał nam jeszcze jeden program (listing 5) wyświetlający wszystkie rekordy.

I w ten sposób założyliśmy najprostszą bazę danych na C-64. Ja również proponuję Wam napisać własne programy obsługujące, umożliwiające wydruk rekordów w porządku alfabetycznym, poszukiwanie wg kodu i miasta itd. - przykłady można by ciągnąć w nieskończoność. Tak więc wiele sukcesów w zmaganiach ze zbiorami typu relative (REL) życzy

MARIUSZ FERDYN

Literatura:

1. Commodore 1541-II Disk Drive user's guide (wydane w USA)
2. Klaudiusz Dybowski, Commodore 64/128 BASIC, SOETO 1987
3. Christian Grzenkowicz, "Najprostsza baza danych dla C-128", Commodore & Amiga 7/92

LISTING 1

```
100 REM *             LISTING #1              *
110 :
120 REM * OTWARCIE ZBIORU REL "ADRESY" *
130 REM *  DLUGOSC REKORDU: 80 ZNAKOW  *
140 P$=CHR$ (13):PRINT CHR$ (147)
150 PRINT CHR$ (18)"NAZWISKO, IMIE:"
160 INPUT A$
170 PRINT CHR$ (18)"ULICA, NR DOMU/NR
    MIESZKANIA:"
180 INPUT B$
190 PRINT CHR$ (18)"KOD, MIASTO:"
200 INPUT C$
210 OPEN 15,8,15
220 OPEN 1,8,2,"ADRESY,L,"+CHR$ (80)
230 PRINT# 15,"P"+CHR$ (96+2)+CHR$ (2)+
    CHR$ (0)+CHR$ (1)
240 PRINT# 1,A$;P$;B$;P$;C$
250 CLOSE 1
260 LA$=STR$ (2)
270 OPEN 1,[illegible],"ADRESY"
280 PRINT# 15,"P"+CHR$ (96+2)+CHR$ (1)+
    CHR$ (0)+CHR$ [illegible]
290 PRINT# 1,[illegible]
300 CLOSE 1[illegible] 15
310 PRINT CHR$ (147):END
```

LISTING 2

```
100 REM *             LISTING #2              *
110 :
120 REM * ZAPIS ADRESOW W Z[illegible] REL *
[illegible]
200 [illegible]
[illegible] PRINT [illegible] (1[illegible]
    MI[illegible]"
[illegible]
230 PRINT [illegible] (18)"KOD, MIASTO:"
240 [illegible]250 OPEN 1,8,2,"ADRESY"
```

JAK NAPISAĆ WŁASNE DEMO

cz. I

LISTING 3

LISTING 4

LISTING 5

NA POCZĄTEK TROCHĘ HISTORII

Seria artykułów „Jak napisać własną grę?" cieszyła się wśród Czytelników ogromną popularnością. W wielu listach przychodzących do redakcji powtarza się pytanie, dlaczego nie publikujemy podobnej serii artykułów dotyczących pisania własnego dema. Otóż tym cyklem postaramy się zadośćuczynić tym prośbom. Dodać należy, że wiele bardzo przydatnych do naszych celów wiadomości można znaleźć w cyklu artykułów Bartłomieja I. Kachniarza z serii „Pamiętnik artylerzysty"

Spróbujmy zastanowić się, co to właściwie jest demo? Zacznijmy od początku. Jak wiadomo Commodore 64 został wyprodukowany jako komputer domowy, nic więc dziwnego, że od razu trafił pod strzechy. Pośród jego użytkowników znalazło się niemało nie mających co robić maniaków. Po opanowaniu umiejętności obsługi nędznego interpretera języka BASIC zapragnęli oni robić rzeczy bardziej ambitne.

Należało do nich przede wszystkim wyciąganie (wypruwanie, po angielsku określane słowem rip) muzyczek z różnego rodzaju gier. Daną muzyczkę umieszczało się w oddzielnym pliku, dodając do niej obrazek zapisany kilkoma słowami dumnego komentarza oraz pozdrowieniami do wszystkich krewnych i znajomych. W ten sposób narodziły się pierwsze dema.

Cóż, nie były one zbyt imponujące, ale jak na tamte czasy było to jednak coś. Jednak ten gatunek, jak wszystko inne, ulegał ewolucji. Początkowo do jednego pliku łączono muzyczki z kilku gier, które wybierało się naciskając odpowiednie klawisze, przykładem może tu być seria demek pt. „SOFT SYNTH (1,2,3,...)". Kolejnym krokiem było urozmaicenie oglądanego obrazu poprzez różnego rodzaju migotanie, podświetlanie napisów, itd. itp. Następnie powstały dema, które zamiast oględnie mówiąc dość średnio wyglądającego ekranu tekstowego zawierały rysunki w trybie graficznym (skądinąd też wypruwane z gierek). Co ambitniejsi dodawali do tego jeszcze *scroll*, czyli mniej lub bardziej płynnie przesuwający się napis, a ci, którym Pan Bóg nie poskąpił talentu, umieszczali nawet własnoręcznie, albo raczej „własnojoystickowo" wykonane prace.

W tym czasie pojawiło się wiele programów ułatwiających tworzenie takich właśnie dem. Różnego rodzaju demo-maker'y umożliwiały wybór muzyczek, rysunków, fontów (czyli kroju liter) i połączenie ich w jedną całość. Co tu dużo mówić – mimo, iż autorzy starali się bardzo uatrakcyjnić swoje produkcje poprzez pisanie dowcipnych, a czasem nawet tragicznie głupich tekstów, ewolucja poszła dalej.

Kolejnym krokiem były SLIDE-SHOW'y (po naszemu pokazy slajdów). Najczęściej były to przenoszone z innych komputerów (IBM, Tandy, Apple) porno obrazki. Pomysł był niegłupi, lecz niestety jakość owych obrazków pozostawiała naprawdę wiele do życzenia.

Korzeni demek można się także doszukiwać w intrach do gier (intro = introduction, po polsku wstęp). Otóż grupki ludzi zajmujących się łamaniem programów, to jest usuwaniem wszelkich zabezpieczeń przed kopiowaniem, przed każdą

Obrazek wykonany w technice FLI Interlace



grą umieszczały swoistą autoreklamę, wyglądającą na ogół w ten sposób: na górze ekranu logo (znak „firmowy" danej grupy; nie mylić z językiem LOGO!), pod spodem tytuł gry, na dole *scroll*. Do najbardziej pamiętnych grup zajmujących się *crack'owaniem* należały Triad, EagleSoft, Radwar, Dynamic Duo, Hotline, Ikari.

Z biegiem czasu intra stawały się coraz bardziej urozmaicone, pojawiało się coraz więcej akcji, wszystko zaczynało się majtać na prawo i lewo, górę i dół, błyskać. Od czasu, kiedy pojawiły się na rynku edytory muzyczne, w intrach zaczęła gościć specjalnie do nich pisana muzyka.

Mniej więcej w tym czasie nastąpił rozwój „sceny komputerowej" takiej, jaką znamy dzisiaj. Nastąpiła dość silna specjalizacja.

Demo „Wonderland 9" grupy CENSOR DESIGN



I tak graficy, jak łatwo się domyśleć, mieli tworzyć rysunki i loga, *swapperzy* „utrzymywać kontakt z innymi grupami, koderzy zajmowali się pisaniem programów, a muzycy – tworzeniem melodii.

W tym samym okresie zaczęto pisać dema, które składały się z kilku następujących po sobie intr, i te możemy już zaliczyć do gatunku, jaki króluje do dzisiaj. Z początku dema te niewiele różniły się od intr, ale z czasem, zwłaszcza od momentu, kiedy pojawił się pierwszy magazyn dyskowy „Sex & Crime" i zaczął podawać listy rankingowe najlepszych twórców na C-64, tempo rywalizacji znacznie wzrosło. W demach zaczęło się pojawiać coraz więcej nowych, ciekawych, a czasem wręcz zaskakujących efektów. Zaczęto robić rzeczy „niemożliwe", takie jak np. wyświetlanie grafiki na ramce. Od tego momentu zaczął występować coraz wyraźniejszy podział pomiędzy *cracker'ami* a ludźmi zajmującymi się demami, chociaż niektóre grupy zachowały obie sekcje.

Gdzieś w międzyczasie rozwinęło się jeszcze jedno zjawisko: DIGITALIZACJA. Można to też zaliczyć do dem. Digitalizacja to po prostu muzyka przetworzona na postać cyfrową i odtwarzana przez komputer. W ten sposób można uzyskać to, że komputer „mówi", „śpiewa", czy odtwarza dźwięki o bardzo realistycznym brzmieniu. W tym gatunku największe sukcesy święciła kiedyś grupa DDG (Digitize Design Group), a teraz digitalizacjami lubi się bawić grupa Censor Design.

Kolejnym przełomowym okresem w historii demek było pojawienie się Amigi. Wiele grup czerpało i do tej pory czerpie inspirację z tej o wiele potężniejszej od C-64 maszyny. Dzięki jej szybkości i systemowi operacyjnemu, tworzenie np. grafiki wektorowej nie jest zbyt trudne. Jednak maniacy sześćdziesiątkiczwórki przeszli samych siebie i dzisiaj prawie każdy szanujący się koder jest w stanie napisać procedurę do obsługi grafiki wektorowej na C-64, a to, co pokazali nam w swym ostatnim demie pt. „The Last Traktor" koderzy z legendarnej już grupy Horizon, robi naprawdę duże wrażenie.

Inną rzeczą, jaką komodorowcy podpatrzyli na Amidze, jest kolejny typ dema. Otóż po pierwszej części pokazuje się loader, którego zadaniem jest uprzyjemnienie czasu podczas ładowania kolejnej części. Demo wygląda w taki sposób: część-loader-część-loader-... ...-loader-część. Następną odmianą, też ściągniętą z Amigi, były dema z menu. Wygląda to na ogół w ten sposób: najpierw oglądamy intro, kolejnym etapem jest menu, z którego wybieramy sobie część, jaką chcemy obejrzeć. Kiedy znudzi nam się dana część, wracamy z powrotem do menu i możemy dokonać kolejnego wyboru. Klasycznym przykładem tego typu dema jest „McDonald Restaurant" grupy Crest.

W ostatnich miesiącach coraz bardziej popularne stają się dema w stylu Amigi. To znaczy przejścia pomiędzy częściami są płynne, a kolejna część ładuje się podczas wykonywania poprzedniej. Jedno z najwybitniejszych dem tego gatunku to „Red Storm" grupy Triad.

Kolejną nowinką rodem z Amigi jest technika zwana *ray-tracing* – są to animacje liczone tak, aby stworzyć wrażenie, że obraz jest rzeczywisty, uwzględnia się tu takie rzeczy jak załamanie i odbicie światła, przez co efekty naprawdę robią wrażenie. Jeszcze inne są dema wzorowane na teledyskach. Chodzi o to, że odtwarzana jest muzyczka, a na ekranie roi się od różnego rodzaju efektów, które wciąż się zmieniają, najczęściej w rytm muzyki. Klasycznym przykładem takiego dema jest „Bonzaied" grupy Bonzai.

Jak widać nie ma jednoznacznej definicji dema. Poza tym bogactwo rodzajów i efektów nie pozwala na łatwe i szybkie nauczenie się jak robić dema. Będziemy to czynić systematycznie, ale należy się przygotować na długą i mozolną pracę, gdyż samo przeczytanie artykułu nie wystarczy, trzeba samemu ćwiczyć, ćwiczyć i jeszcze raz ćwiczyć. Jak mówi stare przysłowie: trening czyni mistrza.

Do zobaczenia w następnym odcinku gdzie zapoznamy się z niezbędnym do dalszej pracy słownictwem.

RAFAŁ PIASEK

ASEMBLER 6502 (cz. VI)

ROZKAZ CMP ET CONSORTES

Czy pamiętacie jeszcze, jak w upalne lipcowe dni wypisywaliśmy na ekranie jakieś dziwne słowa o maniakach i pętlach? Tak? To dobrze. Nie? To nieszczególnie dobrze – powtórne przejrzenie tego odcinka nie zaszkodzi. Co my tu mamy... CPX – instrukcja, która wtedy właśnie pierwszy raz się pojawiła. A służyła... do porównania zawartości rejestru X z podaną liczbą.

Tu wypada postawić pytanie: na jakiej zasadzie rozkaz ten działa? Komputer odejmuje liczbę podaną jako argument (albo też zawartość komórki podanej jako argument, ale to i tak na jedno wychodzi) od zawartości rejestru X. Tu trzeba jeszcze zauważyć, że odejmowanie to odbywa się bez przeniesienia, czyli stan flagi C przed wykonaniem tego rozkazu nie ma wpływu na wynik (czyli inaczej niż w SBC!). Sam wynik nie jest nigdzie przechowywany, ale – żeby całe nasze działanie nie poszło na marne - komputer odpowiednio do wyniku wygasza lub zapala znaczniki: Zera, Przeniesienia i Liczby ujemnej.

Ale jak w praktyce wygląda ta „odpowiedniość"? Najpierw zbadajmy zachowanie znacznika Zero. Wiemy, że jeśli znacznik ten jest zapalony, to oznacza to, że wynikiem ostatniej operacji matematycznej było zero. Jeżeli zaś wynik był różny od zera, to flaga C będzie zgaszona. Wpiszmy więc, tak dla próby, ten oto program:

MONITOR	ASEMBLER
	*=10000
A2710 LDA #$30	LDA #"0"
A2712 LDX #$17	LDX #$17
A2714 CPX #$17	CPX #$17
A2716 BEQ $271a	BEQ PISZ
A2718 LDA #$31	LDA #"1"
A271a STA $0400	PISZ STA $0400
A271d RTS	RTS

Uruchamiamy nasz program i... co widzimy? W lewym górnym rogu ekranu pojawiło się zero. Dlaczego tak się stało? Zanalizujmy program. Najpierw do akumulatora wstawiamy liczbę $30, co odpowiada kodowi ekranowemu zera. Potem do rejestru X wstawiamy liczbę $17, zaraz zaś potem porównujemy zawartość X z liczbą $17 (przy czym sama liczba nie jest tu aż taka istotna; ważne, by była w obu rozkazach taka sama). Po operacji porównania następuje nowy rozkaz: BEQ. BEQ $adres to rozkaz, po wydaniu którego komputer sprawdza stan flagi Zero. Jeśli flaga ta jest zgaszona, to ze spokojem przechodzi do następnego rozkazu. Jeżeli zaś znacznik jest zapalony, to program skacze do komórki o podanym adresie. BEQ to skrót angielskich słów „Branch if EQual to zero", czyli (tłumaczenie wolne) „skocz, jeśli wynik równy był zeru". Oznacza to, że jeśli wynikiem operacji CPX #$17 było 0, to skoczyć trzeba do $271a (oznaczonego etykietą PISZ). Tam znajduje się, znany nam już, rozkaz STA powodujący wydrukowanie zera w lewym górnym rogu ekranu.

Dobrze. Wszystko działa więc zgodnie z przewidywaniami. Zróbmy teraz mały eksperyment: zmień liczbę przy rozkazie CPX na – dajmy na to - $47 albo $00 – byle nie było to $17. Uruchom teraz program. I co? Znowu jest zero? Nie! Tym razem w rogu ekranu pojawiła się jedynka. O czym to świadczy? Przejrzyjmy jeszcze raz program... Najwyraźniej rozkaz BEQ nie został wykonany, przez co do akumulatora wzięta została liczba $31 (oznaczająca jedynkę). Mamy więc pierwsze konstruktywne wnioski: jeżeli argument przy rozkazie CPX równy jest zawartości rejestru X, to zapala się wskaźnik Zero. Jeśli liczby te będą różne, wskaźnik ten gaśnie.

Sprawdźmy teraz, jak – w zależności od argumentów – zachowa się wskaźnik Przeniesienia, czyli C. Zmodyfikujmy nasz program tak, by wyglądał jak ten listing:

MONITOR	ASEMBLER
	*=10000
A2710 LDA #$30	LDA #"0"
A2712 LDX #$17	LDX #$17
A2714 CPX #$15	CPX #$15
A2716 BCC $271a	BCC PISZ
A2718 LDA #$31	LDA #"1"
A271a STA $0400	PISZ STA $0400
A271d RTS	RTS

Przed próbą uruchomienia winien jestem jeszcze wyjaśnienia na temat nieznanego dotąd rozkazu: BCC. BCC to kolejny skok warunkowy. Jego wykonanie zależne jest od stanu znacznika C, czyli Przeniesienia. Jeśli C jest zgaszone, to komputer skacze do miejsca pamięci podanego jako argument rozkazu. Jeśli C jest zapalone, to rozkaz jest ignorowany. Skrót BCC (bo wszystkie mnemoniki asemblera procesora 6502 to TLS-y*, po prostu) oznacza „Branch if Carry Clear", czyli „skocz, jeśli znacznik przeniesienia jest zgaszony". Wszystko jasne?

To jadziem. Wynikiem działania programu jest tym razem jedynka. Oznacza to, że flaga C jest zapalona. Proponuję teraz kilka prób ze zmianą liczby przy rozkazie CPX. (Teraz Ty działasz!) No i co? Masz już gotowe wnioski? Jeśli tak, to porównajmy je z moimi. Mi wyszło, że:

1) Jeśli zawartość rejestru jest większa niż liczba przy rozkazie porównania, to znacznik C będzie zapalony.

2) Jeśli zawartość rejestru jest równa liczbie przy rozkazie porównania, to znacznik C będzie zapalony.

3) Jeśli zawartość rejestru jest mniejsza niż liczba przy rozkazie porównania, to znacznik C będzie zgaszony.

Teraz zmień w linii BCC $271a (czy też BCC PISZ) mnemonik BCC na BCS. Rozkaz BCS oznacza „Branch if Carry Set", czyli „skocz, jeśli znacznik przeniesienia jest zapalony", zaś jego działanie jest dokładnie odwrotne niż BCC. Teraz jeśli uruchomimy program, wyniki będą dokładnie odwrotne niż poprzednio. Wiadomo dlaczego?

Kolejnym znacznikiem, na który wpływać ma rozkaz CPX, jest znacznik Liczby ujemnej. Po zmianie rozkazu skoku program wyglądał będzie tak:

MONITOR	ASEMBLER
	*=10000
A2710 LDA #$2b	LDA #"+"
A2712 LDX #$17	LDX #$17
A2714 CPX #$15	CPX #$15
A2716 BPL $271a	BPL PISZ
A2718 LDA #$2d	LDA #"-"
A271a STA $0400	PISZ STA $0400
A271d RTS	RTS

Nowy rozkaz BPL oznacza „Branch if PLus", czyli „skocz, jeśli dodatnia". Tym razem skok zależeć więc będzie od faktu, czy wynik ostatniej operacji będzie ujemny czy dodatni. W artytmetyce dwójkowej przyjętej przez projektantów układu 6502 (czyli tego procesora, który siedzi w Twoim C-64), liczby ujemne odróżnia się od dodatnich tym, że mają one zapalony siódmy, najstarszy bit. I to właśnie jest najśmieszniejsze. Jeżeli masz np. liczbę %10010011, to nigdy nie wiesz, czy dziesiętnie jest to (128+16+2+1) 147, czy też (-16-2-1) -19. Po prostu nie wiadomo. Dlatego też korzystanie z liczb ujemnych w asemblerze jest, delikatnie mówiąc, ograniczane w miarę możliwości. Zwłaszcza, że w bajcie, którego pierwszy bit jest stracony na znak liczby, mieści się dokładnie dwa razy mniej informacji niż w zwykłym. Pomimo tego wszystkiego ze wskaźnika Liczb ujemnych (Negative) przydaje się korzystać w innych sytuacjach. Przy współpracy z rozkazem CPX zastosowanie jego jest jednak dość ograniczone.

Teoretycznie wskaźnik N zostaje zapalony, gdy zawartość rejestru jest mniejsza od liczby, do której chcemy ją porównać. W rzeczywistości napotykamy jednak kilka problemów. Na przykład, gdybyśmy wzięli liczbę $ff (255) i porównali ją z liczbą $04 (4), to flaga N zostałaby zapalona. Czyli 4 ma być większe od 255? Ale numer! A to wynika po prostu z tego, że liczba 255 jest zbyt duża, by można do niej stosować zasady ujemności. Ujemne liczby muszą się zawierać między -127 a 127. Inaczej wychodzą brednie. Podobne brednie wychodzą i w innych przypadkach. Czyli: do wyników rozkazu BPL powinniśmy zachować dość daleko idącą rezerwę i starać się nie nadużywać tego skoku.

Rozkazem przeciwnym BPL jest BMI ("Branch if MInus", "skocz, jeśli ujemna"). Stosowanie jego napotyka podobne kłopoty jak BPL.

Cierpliwy i uważny Czytelnik zapewne zauważył już, że jakoś dziwnie tekst nie jest powiązany z tytułem. W nagłówku jest bowiem mowa o jakimś CMP, zaś my - jak do tej pory - zajmujemy się tylko tymi "consortes". Czas wreszcie przedstawić głównego bohatera: rozkaz CMP!

CMP oznacza "CoMPare accumulator" czyli "porównaj akumulator". Działanie jego jest identyczne, dokładnie takie samo - prócz jednego szczególika - jak CPX. Szczególikiem tym jest fakt, że jako "mięso armatnie" do porównywania nie idzie zawartość rejestru X, a samego akumulatora. W świecie rozkazu CMP nie może - skoro jest rozkaz CPX odpowiedzialny za rejetr X - zabraknąć rozkazu porównania dla rejestru Y. Rozkazem tym jest CPY ("ComPare Y", "porównaj Y). Analogicznie, wszystkie flagi ustawiane są dokładnie tak samo, jak po rozkazach CMP i CPX.

BARTEK KACHNIARZ

* TLS to skrót od słów TrzyLiterowy Skrót. Zwrotu tego używamy, gdy chcemy zrobić na kimś dobre wrażenie lub zaciemnić nieco sytuację.

UWAGA!
Wszystkie programy z tego artykułu wpisywać można tylko przy pomocy programu MONITOR albo ASEMBLER.

Dziś poznaliśmy rozkazy:

- **CMP** – porównanie zawartości akumulatora z podaną liczbą
- **CPY** – porównanie zawartości rejestru Y z podaną liczbą
- **BEQ** – skok do podanego adresu, pod warunkiem, że wynikiem ostatniej operacji matematycznej było 0.
- **BCC** – skok do podanego adresu, pod warunkiem, że w wyniku osstatniej operacji matematycznej znacznik przeniesienia został zgaszony.
- **BCS** – skok do podanego adresu, pod warunkiem, że w wyniku ostatniej operacji matematycznej znacznik przeniesienia został zapalony.
- **BPL** – skok do podanego adresu, pod warunkiem, że wynikiem ostatniej operacji matematycznej była liczba dodatnia.
- **BMI** – skok do podanego adresu, pod warunkiem, że wynikiem ostatniej operacji matematycznej była liczba ujemna.

FILE CODER

Po ukazaniu się w numerze Commodore & Amiga 03/92 programu Disk Coder, do redakcji nadchodzą listy z zapytaniem, czy można zakodować na hasło nie całą dyskietkę, lecz pojedynczy plik. Do tego właśnie celu służy poniższy program.

Aby zakodować plik należy uruchomić program, następnie wczytać program do zakodowania i wykonać SYS 52736. Teraz trzeba podać hasło. Gdy na ekranie pojawi się kursor, oznacza to, że program jest już zakodowany i można go zapisać na nośniku.

Aby uruchomić zakodowany wcześniej program, należy go po prostu wczytać do pamięci i wykonać RUN. Komputer zapyta teraz o hasło; jeżeli podasz prawidłowe, program zostanie zdekodowany i uruchomiony.

Przypomnę jeszcze, że odnalezienie prawidłowego hasła w sposób przypadkowy jest mało prawdopodobne i wymagałoby - przy haśle złożonym z 5 liter - sprawdzenia ponad 90 milionów kombinacji.

Jeżeli uruchamiasz zakodowany program w obecności osób postronnych, przy wpisywaniu hasła powinieneś zmienić kolor kursora na niebieski (CONTROL - 6) lub dopasowany do koloru ekranu.

UWAGA:
Program do zakodowania nie może być dłuższy niż 196 bloków; ponadto w wypadku ważnych i cennych programów powinieneś zawsze trzymać również w bezpiecznym miejscu ich NIE ZABEZPIECZONE kopie.

MARIUSZ FERDYN

```
200 rem ************************
205 rem *      File coder      *
210 rem *          by          *
215 rem *      M. Ferdyn       *
220 rem ************************
225 d=52736:b=56286
230 c=0:e=d
235 read a$:if a$="end" then 285
240 a1=asc(left$(a$,1)) and 63
245 a2=asc(right$(a$,1)) and 63
250 if a1>47 then 260
255 a1=a1+9:goto 265
260 a1=a1-48
265 if a2>47 then a2=a2-48:
    goto 275
270 a2=a2+9
275 a=a1*16+a2:poke d,a
280 d=d+1:c=c+a:goto 235
285 if c<>b then print "blad w
    li niach data":stop
290 print chr$(147):print "file c
    oder by m. ferdyn"
295 print:print "1. wczytaj program
    do zakodowania."
300 print "2. wykonaj: sys 52736."
305 print "3. gdy pojawi sie kursor
    program
310 print "   bedzie juz zakodowany.
    [illegible]
315 print "   go na nosniku."
320 new
325 data [illegible],a5,01,29,fe,85,01,a2
330 data [illegible],8e,1d,ce,a9,e0,8d,1c
335 data ce,a9,cd,8d,20,ce,a9,ff
340 data 8d,1f,[illegible],ad,00,c0,8d,00
345 data [illegible],[illegible],[illegible],d0,ad,1c,ce,c9
350 data 01,f0,29,ad,1c,ce,c9,00
355 data f0,06,[illegible],[illegible],[illegible],4c,3e,[illegible]
360 data ce,[illegible],[illegible],ce,1d,[illegible],ad,[illegible]
365 data [illegible],c9,[illegible],[illegible],06,ce,1f,ce
370 data [illegible],[illegible],[illegible],ce,1f,[illegible],[illegible],20
375 data [illegible],4c,1b,[illegible],ad,1d,[illegible],[illegible]
380 data [illegible],f0,03,4c,2b,ce,a[illegible],[illegible]
385 data 85,fb,a9,01,85,f[illegible],a[illegible],[illegible]
390 data [illegible],fc,a9,08,85,fe,a0,00
395 data a2,ff,b1,fb,c9,76,f0,10
400 data 91,fd,ee,20,d0,c8,d0,f2
405 data e6,fc,e6,fe,ca,4c,72,ce
410 data a5,01,09,01,85,01,a9,0d
415 data 8d,29,08,a9,60,8d,d4,08
420 data 20,1e,08,a9,93,8d,29,08
425 data a9,a2,8d,d4,08,a5,2d,18
430 data 69,1f,85,2d,a5,2e,18,69
435 data 01,85,2e,a9,bb,a0,ce,20
440 data 1e,ab,60,0d,43,4f,4d,50
445 data 4c,45,54,45,20,21,00,17
450 data 08,00,00,9e,32,30,37,38
455 data 20,42,59,20,4d,2e,46,45
460 data 52,44,59,4e,00,00,00,18
465 data 7d,51,01,60,a2,30,a9,00
470 data 9d,4f,01,ca,d0,fa,a9,93
475 data 20,d2,ff,a9,0d,20,d2,ff
480 data a9,c3,a0,08,20,1e,ab,a2
485 data 00,20,cf,ff,c9,0d,f0,06
490 data 9d,50,01,e8,d0,f3,e0,00
495 data f0,49,8e,d3,08,78,a5,01
500 data 29,fe,85,01,a9,09,8d,68
505 data 08,8d,6e,08,a9,20,8d,67
510 data 08,8d,6d,08,ad,01,08,20
515 data 9f,08,8d,01,08,ad,67,08
520 data c9,ff,f0,09,ee,67,08,ee
525 data 6d,08,4c,66,08,ad,68,08
530 data c9,cd,f0,0f,ee,67,08,ee
535 data 68,08,ee,6d,08,ee,6e,08
540 data 4c,66,08,a5,01,09,01,85
545 data 01,58,4c,d4,08,a0,00,59
550 data 50,01,a2,00,85,02,bd,50
555 data 01,20,19,08,9d,50,01,e8
560 data ec,d3,08,d0,f1,a5,02,c8
565 data cc,d3,08,d0,e2,ee,20,d0
570 data 60,57,50,52,4f,57,41,44
575 data 5a,20,48,41,53,4c,4f,3a
580 data 00,00,a2,00,bd,e7,08,9d
585 data 20,01,c9,a7,f0,04,e8,4c
590 data d6,08,4c,20,01,78,a5,01
595 data 29,fe,85,01,a9,20,85,fb
600 data a9,01,85,fd,a9,09,85,fc
605 data a9,08,85,fe,a0,00,a2,c7
610 data b1,fb,91,fd,ce,20,d0,c8
615 data d0,f6,e6,fc,e6,fe,ca,d0
620 data ef,a5,01,09,01,85,01,58
625 data 20,59,a6,4c,ae,a7,76,end
```

„RZUTY"

CAŁA PRAWDA O FUNKCJI BALISTYCZNEJ



Jeśli ktokolwiek buduje własną armatę i chciałby wyliczyć, jaki będzie jej zasięg rażenia, powinien skorzystać z tego programu... A mówiąc serio: „Rzuty" pozwolą Ci zobaczyć na własne oczy przebieg funkcji balistycznej, czyli zależność odległości końcowej od kąta wyrzutu i prędkości początkowej ciała. Sądzę, że jedna czy dwie godzinki spędzone sam na sam z programem zagwarantują Ci pełne zrozumienie sprawy.

UWAGA!
Program napisany jest w SIMON'S BASIC'u i nie uwzględnia w obliczeniach masy wyrzucanego ciała. Można natomiast zmieniać wysokość początkową. Ze względu na dużą ilość czasu zużywanego na obliczenia zrezygnowałem też z rysowania linii ciągłych symbolizujących trajektorię lotu - zamiast nich wyświetlane są w niewielkich odstępach pojedyncze punkty, co w zupełności wystarcza.

Życzę dobrej zabawy.

WOJCIECH BENBENEK

```
100 rem *****************
110 rem *    "Rzuty"    *
120 rem *     autor     *
130 rem *     W.B.E.    *
140 rem *****************
150 :
160 hires6,12:colour6,12
170 text100,60,"Rzuty",1,5,20
180 text200,190,"autor W.B.E.",1,1,8
190 text60,150,"(wcisnij dowolny klawisz)",1,1,8
200 poke198,0:wait198,1
210 pause1:nrm:clr
220 printchr$(147)
230 input"predkosc poczatkowa (40) v=";v:ifv=0thenv=40
240 input"wysokosc poczatkowa (0)  h=";h:z=15
250 ifv+h>60then220
260 ifh>0thenz=0
270 hires7,15
280 multi2,0,6:colour12,12
290 block0,0,140,160,3:h$="wys.pocz.":v$
    ="predk.pocz.":block60,0,140,8,0:r=0
300 block74,8,140,17,0:block100,17,140,26,0:k$="kat"
310 fora=zto75step15
320 g=9.8123:od$="odleglosc":f$=str$(int
    (x)):us$="     ":rem 5 spacji
330 text75,10,h$+str$(int(h)),3,1,7:text
    61,0,v$+str$(v),3,1,7
340 s$=" ":rem 1 spacja
350 text103,19,k$+s$+s$+str$(a),2,1,7
360 fort=0to10step.1
370 x=v*t*cos(a*Π/180):y=h+v*t*sin(a*Π/180)-g*t^2/2
380 text0,170,od$+us$+str$(int(x)),2,1,7
    :text0,170,od$+us$+str$(int(x)),0,1,7
390 c=160-(2*y):ifc<0then420
400 ify<=0andt>0then430
410 plotx,c,4
420 next:next
430 am$=" max"
440 text0,190,od$+am$+" ,1'"+f$,0,1,7:te
    xt103,19,k$+s$+s$+str$(a),0,1,7
450 text0,180,k$+us$+us$+" ,a'"+str$(a-15),0,1,7:gosub490
460 text0,190,od$+am$+" ,1'"+str$(int(x)),1,1,7
470 text0,180,k$+us$+us$+" ,a'"+str$(a),3,1,7
480 t=0:pause4:nexta:goto550
490 ifa=15thenw1$=str$(int(x))
500 ifa=30thenw2$=str$(int(x))
510 ifa=45thenw3$=str$(int(x))
520 ifa=60thenw4$=str$(int(x))
530 ifa=75thenw5$=str$(int(x))
540 return
550 text0,190,od$+am$+" ,1'"+str$(int(x)),0,1,7
560 text0,180,k$+us$+us$+" ,a'"+str$(a-15),0,1,7
570 sq=sqr(v^2+2*g*h)
580 am=atn(v/sq)*180/Π
590 lm=v*sq/g
600 a1$=str$(int(am))
610 lm$=str$(int(lm))
620 text0,170,od$+am$+lm$+lm$+s$+"przy",2,1,7
630 text0,185,k$+us$+a1$,2,1,7
640 text103,19,ks$+s$+s$+str$(a-15),3,1,7
650 poke198,0:wait198,1
660 clr:goto210
```

FAST FORMAT

Jak wiadomo, stacje dysków do Commodore 64 nie należą do najszybszych. Formatowanie dyskietki trwa około minuty i dziesięciu sekund. Oczywiście można skorzystać ■ modułu np. Action Replay, Final III. Jednak nawet ■■ ich pomocą sformatowanie 10 dyskietek (czyli dwudziestu stron) może przyprawić co najmniej ■ ból głowy.

Proponuję zatem skorzystać ■ poniższego programu, służącego do szybkiego formatowania dyskietek trwającego około 10 sekund. Oprócz rewelacyjnie krótkiego czasu formatowania program daje możliwość nadania dyskietce pięcioznakowego identyfikatora (tradycyjnie - tylko dwa znaki).

Sama procedura szybkiego formatowania pochodzi ■ programu DIRMASTER V3.1 napisanego przez RUZSA BALAZS (CELLUX) ■ grupy FACES.

MARIUSZ FERDYN

OD REDAKCJI:
Przyspieszenie procesu formatowania wiąże się ■■ zrezygnowaniem ■ weryfikacji (działa tak większość „przyspieszaczy"). Jeśli więc masz dyskietki, które podejrzewasz ■ uszkodzenia, postaraj się sformatować je jednak w "naturalny" sposób.

```
200 rem ************************
205 rem *       Fast Format     *
210 rem *                       *
215 rem * (c) 1991 Ruzsa Balazs *
220 rem *        (Cellux/Faces) *
225 rem *                       *
230 rem *  (c) 1992 M. Ferdyn   *
235 rem ************************
240 print chr$ (147)
245 print "                fast for
    mat":rem 14 spacji
250 print "                    by":
    rem 18 spacji
255 print "               m.ferdy
    n":rem 15 spacji
260 print:print "          fast f
    ormat routine":rem 10 spacji
265 print "                  from"
    :rem 17 spacji
270 print "            dirmaster
    v3.1":rem 12 spacji
275 print "                    by":
    rem 18 spacji
280 print "      ruzsa balazs (ce
    llux/faces)":rem 6 spacji
285 print:print:print "prosze cze
    kac";
290 d=9385:b=76562
295 c=0:e=d
300 read a$:if a$="end" then 350
305 a1=asc (left$ (a$,1)) and 63
310 a2=asc (right$ (a$,1)) and 63
315 if a1>47 then 325
320 a1=a1+9:goto 330
325 a1=a1-48
330 if a2>47 then a2=a2-48:goto 3
    40
335 a2=a2+9
340 a=a1*16+a2:poke d,a:if a=  th
    en print ".";
345 d=d+1:c=c+a:goto 300
350 if c<>b then print "blad ■ li
    niach data":stop
355 clr
360 for a=0 to 15:b=peek (10010+a
    ):a$=a$+chr$(b):next ■
365 print:print
370 print chr$ (17):print "
          *****************":rem
    14 spacji
375 print chr$(145):chr$(145);chr
    $(145)
380 print "nazwa dysku    ";
385 print a$;:for a=1 to 18:print
     chr$(157);:next a
390 input a$:print:print
395 for a=  to 15:poke (10010+a),
    32:next ■
    b=len(a$):if b>16 then b=16
405 for a=1 to b:poke (10009+a),a
    sc(mid$(a$,a,1)):next a
    clr
415 for a=0 to 4:b=peek(10028+a)
    a$=a$+chr$(b):next a
420 print
425 print chr$(17):print "
      ******":rem 11 spacji
430 print chr$(145):chr$(145):chr
    $(145)
435 print "id dysku    ";
    print a$;:for a=1 to 7:print
    chr$(157);:next a
445 input a$:print:print
4   for a=1 to 4:poke (10028+a),3
    2:next a
4   b=len(a$):if b>5 then b=5
    for a=1 to b:poke (10027+a),a
    sc(mid$(a$,a,1)):next a
465 print:print "wl z dysk do sfo
    rmatowania"
    print "i nacisnij ";chr$(18);
    "return";chr$(146);"."
47  get a$:if a$<>chr$(13)goto475
48  sys 9385:rem * formatowanie *
4   open 15,8,15:input# 15,en,em$
    ,et,es
490 print:print en;",";em$:",";et
    ;",";es:close 15
495 print:print ;chr$(18);"f";chr
    $(146);"formatowac nastepna d
    yskietke,"
500 print ;chr$(18);"z";chr$(146)
    ;"zmiana nazwy, "
505 print ;chr$(18);"w";chr$(146)
    ;"powrot do BASIC"
510 get a$:if a$="f" then run 465
515 if a$="z" then run 355
520 if a$="w" then new
525 goto 510
530 rem ************************
535 rem *     kod maszynowy     *
540 rem ************************
545 data a9,00,85,fc,a9,03,85,fd
550 data a9,5b,85,fe,a9,25,85,ff
555 data a9,57,20,42,25,a5,fc,20
560 data a8,ff,a5,fd,20,a8,ff,a9
565 data 20,20,a8,ff,a0,00,b1,fe
570 data 20,a8,ff,c8,c0,20,d0,f6
575 data 20,ae,ff,a5,fc,18,69,20
580 data 85,fc,a5,fd,69,00,85,fd
585 data a5,fe,18,69,20,85,fe,a5
590 data ff,69,00,85,ff,a5,fd,c9
595 data 05,d0,bd,a9,45,20,42,25
600 data a9,00,20,a8,ff,a9,03,20
605 data a8,ff,20,ae,ff,a9,07,8d
610 data 00,dd,2c,00,dd,50,fb,58
615 data a9,08,20,b1,ff,a9,6f,20
620 data 93,ff,a9,55,20,a8,ff,a9
625 data 39,20,a8,ff,20,ae,ff,a9
630 data 08,20,b1,ff,a9,6f,20,93
635 data ff,a9,49,20,a8,ff,4c,ae
640 data ff,48,a9,08,20,b1,ff,a9
645 data 6f,20,93,ff,a9,4d,20,a8
650 data ff,a9,2d,20,a8,ff,68,4c
655 data a8,ff,a5,22,d0,0d,a9,c0
660 data 85,00,58,a5,00,30,fc,24
665 data 20,70,fc,78,ad,00,1c,09
670 data 0c,8d,00,1c,a2,00,88,d0
675 data fd,ca,d0,fa,a9,01,85,18
680 data ad,d4,04,85,16,85,12,ad
685 data d5,04,85,17,85,13,a9,05
690 data 85,31,20,aa,04,a9,00,85
695 data 3a,20,8f,f7,a0,bb,b9,00
700 data 01,99,00,06,c8,d0,f7,a9
705 data 07,85,31,85,6e,a9,84,85
710 data b4,a9,f0,8d,4f,02,20,aa
715 data 04,20,b7,ee,a0,1b,b9,bf
720 data 04,99,90,07,88,10,f7,a9
725 data 41,8d,02,07,a9,2a,8d,03
730 data 07,a9,11,8d,48,07,a9,fc
735 data 8d,49,07,20,e9,f5,85,3a
740 data 20,8f,f7,a5,18,20,69,04
745 data a5,22,c9,12,d0,09,a9,07
750 data 85,31,a9,01,8d,17,04,a9
755 data ff,8d,03,1c,8d,01,1c,a9
760 data ce,8d,0c,1c,a0,00,84,19
765 data 84,c2,84,c0,a5,19,45,18
770 data 45,17,45,16,85,1a,20,34
775 data f9,a2,00,a4,c0,b7,24,99
780 data 00,06,c8,e8,e0,04,d0,f5
785 data e6,19,a5,19,c5,43,90,da
790 data a2,05,50,fe,b8,a9,ff,8d
795 data 01,1c,ca,d0,f5,a2,08,a4
800 data c2,50,fe,b8,b9,00,06,8d
805 data 01,1c,c8,ca,d0,f3,84,c2
810 data a2,0b,50,fe,b8,a9,55,8d
815 data 01,1c,ca,d0,f5,a2,05,50
    data fe,b8,a9,ff,8d,01,1c,ca
825 data d0,f5,a0,bb,50,fe,b8,b9
830 data 00,06,8d,01,1c,c8,d0,f4
835 data 50,fe,b8,b1,30,8d,01,1c
840 data c8,d0,f5,a2,09,50,fe,b8
845 data a9,55,8d,01,1c,ca,d0,f5
850 data a9,05,85,31,a9,06,8d,17
855 data 04,c6,19,d0,93,20,00,fe
860 data a5,18,c9,23,f0,05,e6,18
865 data 4c,e3,03,a9,07,85,31,20
870 data f2,f5,a9,30,85,20,a9,ff
875 data 85,48,a9,00,85,3e,8d,00
880 data 18,58,60,48,38,e5,22,f0
885 data 14,0a,85,4a,20,2e,fa,a9
890 data 95,8d,05,18,ad,05,18,30
895 data fb,a5,4a,d0,ef,68,85,22
900 data a2,04,dd,b2,04,ca,b0,fa
905 data bd,b7,04,85,43,ad,00,1c
910 data 09,08,29,9f,1d,bb,04,8d
915 data 00,1c,a9,ee,8d,0c,1c,a9
920 data 05,85,31,60,a0,00,98,91
925 data 30,c8,d0,fb,60,29,1f,19
930 data 12,11,12,13,15,00,20,40
935 data 60,43,4f,4d,4d,4f,44,4f
940 data 52,45,20,20,20,43,2d,36
945 data 34,a0,a0,44,49,53,4b,31
    ,end
```

ZEGAR DLA C-64

Program ten wyposaża Twój komputer w zegar. Czas jest wyświetlany ■ pierwszej linii ekranu. Wyświetlanie można wyłączyć przez wciśnięcie klawiszy RUN/STOP I RESTORE. Wznowienie wyświetlania uzyskasz po wykonaniu SYS 49152.

Po wpisaniu, ale PRZED uruchomieniem, program należy zapisać na kasecie lub dyskietce. Wpisz RUN i wciśnij klawisz RETURN. Podaj aktualny czas i uruchom zegar wciśnięciem dowolnego klawisza.

SYLWESTER ZIELIŃSKI

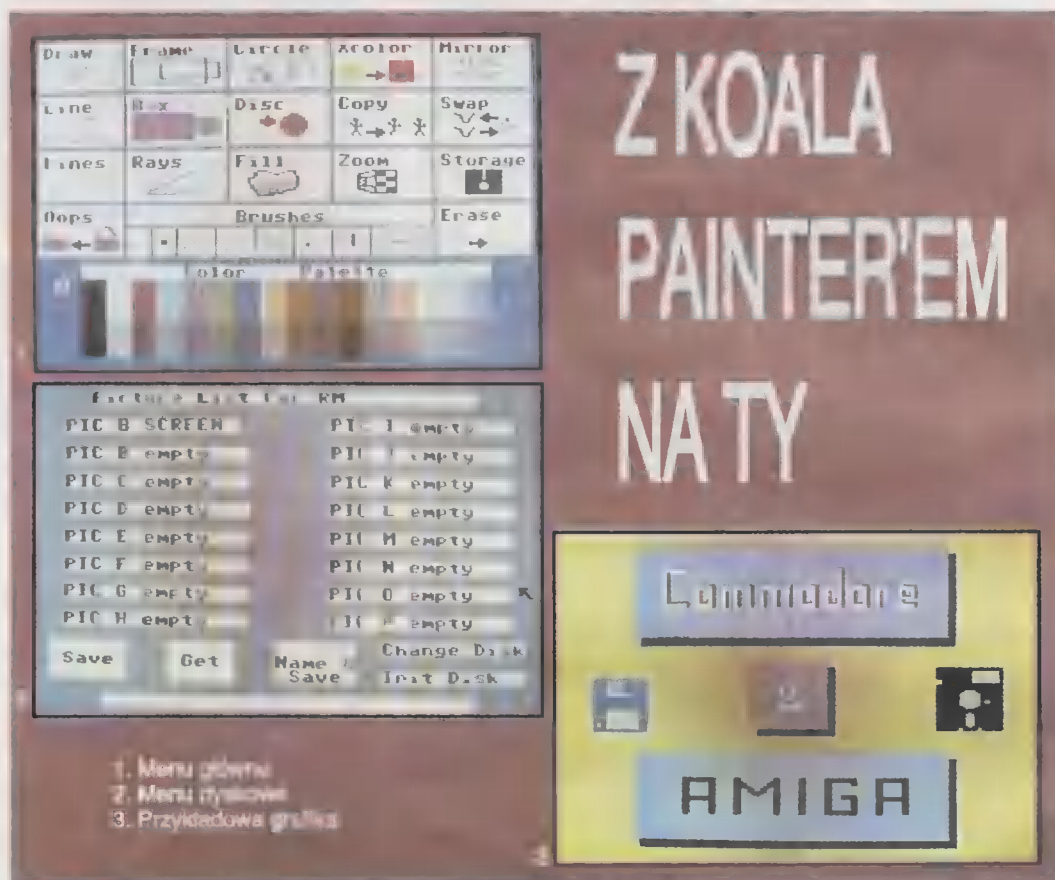
```
100 rem ****************
105 rem *     Zegar     *
110 rem *               *
115 rem *  S. Zielinski *
120 rem ****************
125 :
130 print chr$(147)
135 ad=49152:li=10046
140 b$="end data"
145 c$="blad w danych"
150 d$=chr$(13)
155 :
160 read a$
165 if a$=b$ then 220
170 a1=asc(left$(a$,1))
175 a2=asc(right$(a$,1))
180 if a1>57 then a1=a1-55:goto 190
185 a1=a1-48
190 if a2>57 then a2=a2-55:goto 200
195 a2=a2-48
200 da=a1*16+a2
205 poke ad,da
210 ad=ad+1:li=li-da
215 goto 160
220 if li<>0 then print d$;c$
225 print d$"program wpisany":d$
230 :
235 rem ustawienie czasu
240 input "godzina";h$
245 if h$=" " then 240
250 h=0:if len(h$)>1 then h=16
255 hh=val(right$(h$,1))
260 h=h+hh:poke 56331,h
265 input "minuty";m$
270 if m$=" " then 260
275 m=0:if len(m$)>1 then m=16*v
    al(left$(m$,1))
280 mm=val(right$(m$,1))
285 m=m+mm:poke 56330,m
290 print d$;"nacisnij klawisz ■
    by zegar ruszyl"
295 get a$:if a$="" then 295
300 poke 56329,1:sys 49152
305 print d$;"run stop+restore z
    atrzymuje wyswietlanie"
310 print "sys 49152 je wznawia"
315 new
320 :
325 rem dane kodu maszynowego
330 data 78,a9,0d,8d,14,03,a9,c0
335 data 8d,15,03,58,60,ad,0b,dc
340 data 29,10,4a,4a,4a,4a,69,30
345 data 8d,10,04,ad,0b,dc,29,0f
350 data 69,30,8d,11,04,ad,0a,dc
355 data 29,f0,4a,4a,4a,4a,69,30
360 data 8d,13,04,ad,0a,dc,29,0f
365 data 69,30,8d,14,04,ad,09,dc
370 data 29,f0,4a,4a,4a,4a,69,30
375 data 8d,16,04,ad,09,dc,29,0f
380 data 69,30,8d,17,04,ad,08,dc
385 data 69,30,8d,19,04,a9,20,8d
390 data 0a,04,8d,0f,04,8d,1a,04
395 data a9,3a,8d,12,04,8d,15,04
400 data 8d,18,04,a9,0f,8d,0b,04
405 data a9,0f,8d,0c,04,a9,15,8d
410 data 0d,04,a9,12,8d,0e,04,ad
415 data 09,dc,cd,00,c1,f0,10,ad
420 data 08,dc,ad,09,dc,8d,00,c1
425 data 29,01,f0,03,ee,00,dc,4c
430 data 31,ea,00,00,00,00,00,00
435 ::::::::::::::::::::::::::::
440 data "end data"
```

Jednym z lepszych edytorów graficznych na C-64 jest KOALA PAINTER. Dzięki niemu użytkownicy pragnący tworzyć obrazy graficzne nie muszą główkować nad poleceniami PRINT, POKE itp. Obrazkowe menu pozwala na wybranie dowolnej funkcji bez dotykania klawiatury. Posługujemy się joystickiem lub myszą. KOALA PAINTER umożliwia tworzenie grafiki w trybie multicolor z rozdzielczością 160 na 190 punktów.

Po wczytaniu i uruchomieniu programu ukazuje się główne menu. Składa się ono z następujących ikon:

• DRAW
Przytrzymując przycisk FIRE przesuwamy myszką strzałkę rysując w ten sposób własne znaki, linie itd.

• FRAME
Tworzenie ramek (czworoboków) o różnych kształtach. Obieramy strzałką miejsce, w którym przez jednokrotne naciśnięcie przycisku FIRE zaczepiamy jeden z wierzchołków naszej ramki. Przesuwając myszkę rozciągamy czworobok do interesującej nas wielkości. Ponowne naciśnięcie FIRE kończy proces formowania wielkości i kształtu ramki. Teraz myszką przesuwamy całą ramkę w interesujące nas miejsce na palecie do rysowania. Po raz ostatni naciskamy FIRE i komputer kreśli nasz czworobok.



1. Menu główne
2. Menu [illegible]
3. Przykładowa grafika

• CIRCLE
Kreślenie okręgów o różnych średnicach. Postępujemy w identyczny sposób jak przy funkcji FRAME. Jedyną różnicą jest to, że na początku nie obieramy wierzchołka lecz środek okręgu.

• LINE
Tworzenie linii, których początki i końce są od siebie niezależne. Całą operację kreślenia linii dokonujemy w identyczny sposób jak przy funkcji FRAME.

• LINES
Tworzenie linii zależnych od siebie. Zasada kreślenia jest następująca: strzałką obieramy punkt, w którym znajdzie się początek linii, następnie naciskamy przycisk FIRE i przystępujemy do kształtowania jej. Po ponownym naciśnięciu FIRE komputer wykreśli nam naszą pierwszą linię. Koniec pierwszej linii staje się początkiem drugiej itd. Tworzymy jakby jedną długą łamaną linię.

• RAYS
Tworzenie linii zależnych od siebie. Kreślenia tych linii dokonujemy w ten sposób, iż strzałką obieramy dowolny punkt, w którym znajdzie się jeden wspólny początek dla wszystkich kreślonych linii. Naciskamy FIRE i rozciągamy pierwszą linię. Po ponownym naciśnięciu FIRE komputer wykreśli zaprojektowaną już linię. Początkiem drugiej i następnych linii będzie początek pierwszej.

• BOX
Tworzenie ramek, które następnie wypełniają się obranym przez nas kolorem. Postępujemy w identyczny sposób jak przy funkcji FRAME.

• DISC
Tworzenie okręgów, które wypełniają się obranym przez nas kolorem. Postępujemy tak jak przy funkcji CIRCLE.

• FILL
Wypełnianie wolnych przestrzeni, a także obiektów, obranym przez nas kolorem.

• MIRROR
Pozwala nam na jednoczesne tworzenie czterech takich samych obiektów, np. okręgów. Każdy z nich będzie umieszczony w innej ćwiartce ekranu.

• ZOOM
To polecenie daje nam możliwość dokładnego obejrzenia w dość dużym przybliżeniu wybranych części rysunku. Możemy dokonywać poprawek, zmian itd.

• COPY
Funkcja dająca nam możliwość powielania stworzonego już obiektu. Za pomocą myszki należy rozciągnąć ramkę naokoło interesującego nas obiektu (rozciągać jak przy funkcji FRAME). Po ponownym naciśnięciu FIRE przesuwamy ramkę w odpowiadające nam miejsce na ekranie. Naciskamy FIRE i komputer kreśli obszar, który został objęty ramką. Liczba kopii danego obszaru jest dowolna.

• X COLOR
Zmiana koloru podkładu na którym rysujemy. Umieścić strzałkę na podkładzie i nacisnąć FIRE. Kolor podkładu zmieni się na taki, jaki wcześniej obierzemy.

• SWAP
Funkcja ta pozwala nam przemieścić się do drugiego podkładu, na którym możemy tworzyć drugi rysunek.

• OOPS
Naciśnięcie przycisku FIRE w chwili, gdy strzałka znajduje się na tej ikonie, spowoduje zlikwidowanie ostatnio stworzonego elementu naszego rysunku.

• ERASE
Działa tak jak funkcja OOPS, lecz likwidacja dotyczy całego rysunku.

• STORAGE
Przejście do drugiego menu, które służy do obsługi stacji dysków.

• BRUSHES
Ikony ze znakami wzorów, dzięki którym możemy tworzyć linie, ramki itd.

• COLOR PALETTE
Paleta kolorów podzielona na dwie części. W sumie do rysowania i wypełniania mamy 16 kolorów i wiele możliwości nakładania na te kolory (dolna część palety) siatek o różnych barwach.

Przedostanie się z głównego menu na podkład do kreślenia nie jest żadnym problemem. Należy bowiem strzałką najechać na górną lub dolną ramkę tak, aby nie widać było strzałki. Przerywany sygnał dźwiękowy jest znakiem gotowości komputera do przeniesienia, które nastąpi z chwilą naciśnięcia przycisku FIRE. Powrotu do głównego menu dokonujemy w ten sam sposób.

Jeżeli chcemy przenieść się na drugi podkład kreślarski, to musimy posłużyć się funkcją SWAP. Z głównego menu przenosimy się na pierwszy podkład, a dopiero potem na drugi. To ostatnie przeniesienie wygląda jednak nieco inaczej, niż poprzednie. Znajdujemy się na pierwszym podkładzie, gdzie przy strzałce wyświetlona jest wcześniej uaktywniona funkcja SWAP. Teraz naciskamy FIRE i zostajemy przeniesieni na drugi podkład.

Poznajmy teraz drugie menu - STORAGE. Po wywołaniu tej funkcji komputer odczytuje z dyskietki nazwy wcześniej zapisanych na niej rysunków. Okna o treści PIC A EMPTY są niczym innym jak wykazem nazw 16 rysunków (od PIC A do PIC P). W miejscach EMPTY powinny pojawić się nazwy, jeśli tylko na dyskietce znajdują się pliki z rysunkami.

W menu STORAGE możemy wyodrębnić następujące okna:

▪ **SAVE** Zgranie rysunku na dyskietkę, bez podawania nazwy. Ustawiamy strzałkę na oknie SAVE i naciskamy przycisk FIRE. Przy strzałce powinno pojawić się słowo SAVE. Następnie przesuwamy ją na jedno z wolnych od nazwy okien PIC i ponownie naciskamy FIRE. Przykład wolnego okna PIC - PIC C EMPTY.

• GET
Wczytanie rysunku z dyskietki. Ustawiamy strzałkę na oknie GET i naciskamy przycisk FIRE. Przy strzałce powinno pojawić się słowo GET. Następnie przesuwamy ją na jedno z okien PIC, które ma wpisaną nazwę rysunku i ponownie naciskamy FIRE. Przykład okna PIC posiadającego nazwę - PIC B DIODA Z.

▪ NAME/SAVE
Zgranie rysunku na dyskietkę z podaniem jego nazwy. Ustawiamy strzałkę na oknie NAME/SAVE i naciskamy przycisk FIRE. Przy strzałce powinno pojawić się NAME/SAVE. Następnie przesuwamy ją na jedno z wolnych od nazwy okien PIC i ponownie naciskamy FIRE. Teraz z klawiatury wpisujemy nazwę naszego dzieła i naciskamy RETURN.

▪ CHANGE DISC
Czytanie katalogu dyskietki ze zbiorem rysunków. Ustawiamy strzałkę na oknie CHANGE DISC i naciskamy przycisk FIRE. Komputer wyświetla nam nazwy rysunków w 16 oknach PIC.

• INIT DISC
Formatowanie dyskietki wraz z nadaniem jej nowej nazwy. Ustawiamy strzałkę na oknie INIT DISC i naciskamy przycisk FIRE. Komputer prosi nas o potwierdzenie obranej funkcji (Y), po czym wpisujemy nazwę naszego rysunku i naciskamy RETURN.

Obsługa stacji dysków za pomocą menu STORAGE jest dość prosta, nie powinna więc przysporzyć nam większych kłopotów. Powrotu do głównego menu dokonuje się w taki sam sposób jak przeniesienia na podkład kreślarski.

KOALA PAINTER jest w sumie niezłym programem graficznym, choć brak mu wielu takich funkcji, jakie znalazły się w nowszych programach tego typu (np. ART STUDIO). Przykładem może tu być chociażby PRINT (wydruk zaprojektowanego rysunku), czy TEXT (wpisywanie tekstu z klawiatury). Nie należy się jednak tym zrażać, bowiem edytor KOALA PAINTER jest wart poznania, choćby ze względu na jego prostą obsługę. Niech żyją dinozaury!!!

ROBERT KULIŚ

sposoby i sposobiki

zostań włamywaczem!

cz. IV (ostatnia)

GRY DYSKIETKOWE

Formalnie rzecz biorąc „życie", „energia", „czas" itd. w grach dyskietkowych kodowane są identycznie jak w grach taśmowych. Różnica polega na sposobie restartu i dostępu do odszukanych usprawnień. Można przyjąć dwie metody postępowania:

a) po uruchomieniu gry dyskietkowej przechodzimy do monitora, odszukujemy i modyfikujemy interesujące nas fragmenty programu i wreszcie uruchamiamy grę przy pomocy Action Replay'a MK V;
b) wprowadzamy na stałe NA DYSKIETKĘ niezbędne modyfikacje.

Metoda pierwsza pozwala nam oszczędzić (w sensie dosłownym) na dyskietkach, natomiast jej wadą jest konieczność każdorazowego wpisywania niezbędnych poprawek do gry, gdy chcemy zagrać w jej „nieśmiertelną" wersję.

Metoda druga na odwrót: jej zaletą jest natychmiastowy dostęp do „nieśmiertelnej" wersji gry okupiony jednak podwójną liczbą dyskietek, jaką musimy na nią poświęcić. Ponadto odszukiwanie i modyfikacja odpowiednich ciągów bajtów NA DYSKIETCE wymaga posiadania specjalnego programu narzędziowego oraz umiejętności posługiwania się monitorem dyskowym, w jaki wyposażony jest moduł. Za to nie trzeba posiadać Action Replay'a MK V.

Program do poszukiwania na dyskietce zadanej sekwencji bajtów był opisany w „Bajtku" nr 2/91. Działa on na zasadzie wgrywania do określonego obszaru pamięci (CF00-CFFF dla Fina'a II) zawartości dowolnie wybranego sektora dyskietki. Ten obszar pamięci można dis-asemblować, przeszukiwać i modyfikować używając odpowiednich opcji monitora. Zmodyfikowany sektor wgrywamy w odpowiadające mu na dyskietce, właściwe miejsce. Operacji odczytu i zapisu dokonujemy za pomocą rozkazów monitora dyskowego. I tak *R 1A 07 i RETURN odczyta sektor 7 z 26 ścieżki i jego zawartość wpisze do pamięci o adresach CF00 do CFFF. Z kolei *W 1A 07 spowoduje zapis zawartości pamięci o adresach j/w do 7 sektora na 26 ścieżce.

CO DALEJ?

Warto chyba jeszcze wspomnieć o możliwości nieco innego spojrzenia na techniczną stronę poszukiwania interesujących nas informacji. W tym celu trzeba zapoznać się z mapą pamięci C-64. Niektóre jej obszary to po prostu specjalizowane rejestry zajmujące się obsługą określonych urządzeń peryferyjnych, dźwiękiem, duszkami, wykrywaniem ich kolizji itp.

Ponieważ bardzo często naciśnięciu przycisku FIRE joysticka towarzyszy użycie np. amunicji z jednoczesnym zmniejszeniem jej zapasu, to możemy fakt ten wykorzystać do odnalezienia „wiecznej" amunicji, zwłaszcza wtedy, gdy nie dały efektu metody opisywane wcześniej. Zainteresowani powinni wiedzieć, że mikroprocesor 6502 (6510) komunikuje się z joystickami, wiosełkami i klawiaturą za pośrednictwem uniwersalnych portów wejścia/wyjścia układu CIA1 (Complex Interface Adapter 1). Do odczytu informacji z joysticka nr 2 używany jest port A, czyli rejestr $DC00. Stan czwartego bitu tego rejestru określa, czy przycisk FIRE jest wciśnięty. Ta informacja wystarcza do odszukania „wiecznej" amunicji. Jest chyba oczywiste, że musimy zacząć od przeszukania pamięci i znalezienia adresów, pod którymi występuje sekwencja bajtów 00 DC. Pozwoli to nam na zlokalizowanie podprogramu zajmującego się odczytem stanu joysticka. Następnie ustalamy, skąd w pamięci podprogram ten jest wywoływany i analizujemy tak znalezione kolejne podprogramy pod kątem występowania w nich rozkazów DEC i SBC, zwracając także uwagę na ich indeksowane tryby. Najczęściej nadzieje nasze zostają spełnione, chociaż czasem liczba podprogramów, które są wywoływane przez podprogramy, które... itd. rośnie lawinowo i łatwo jest coś przeoczyć.

Naturalnie kilka zaprezentowanych przykładów oraz ogólnie jedynie omówione zagadnienie poszukiwania „usprawnień" do gier dyskietkowych nie wyczerpuje tematu „włamań" do programów. Celem artykułu jest zachęcenie Czytelnika do podjęcia samodzielnej pracy i dlatego stosunkowo dużo miejsca poświęciłem adresowi startowemu i licznikom błędów. Wspomniane zagadnienia są fundamentem działalności „włamywacza".

Omówienie całej reszty przykładów, metod, trików i sposobów przekracza ramy tego opracowania i sprowadziłoby się do przepisywania książki poświęconej tym zagadnieniom, którą napisałem. Ukaże się ona niebawem nakładem SOETO i, jak mam nadzieję, udzieli odpowiedzi na wszelkie pytania i wątpliwości nurtujące początkujących „włamywaczy", a nieco bardziej oblatanym w temacie Czytelnikom powinna wskazać drogę do poszukiwania własnych sposobów na „rozpracowanie" gier.

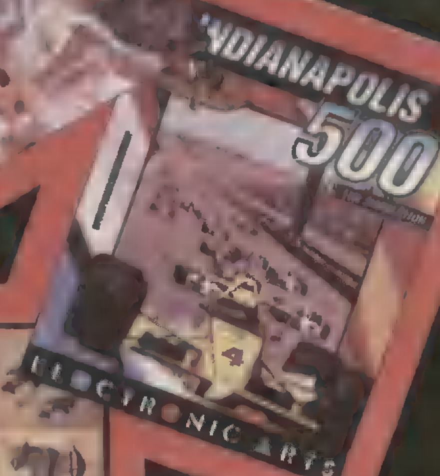
Mam nadzieję, że zachęciłem chociaż część z Was do podjęcia próby nauki asemblera w ten nietypowy sposób.

URAN

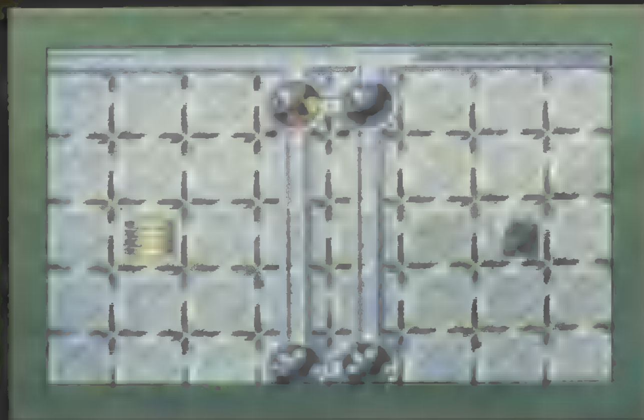
Z bogatej oferty IPS-u







GRY


Logical

PRZEWODNIK PO GRACH - AMIGA

Na początek coś dla tych, którzy lubią sobie trochę pochodzić, poskakać, pofruwać lub popływać. Oczywiście chodzi tu o gry ADVENTURE-ARCADE (przygodowe, zręcznościowe, tekstówki, platformówki), w których Twoim celem jest najczęściej albo uwolnić małe dzieciątko spod wrogich szponów złego czarownika, albo zabić złego czarownika, czy też pozbierać wszystkie skarby z zamku lub też (nawet!) odnaleźć własną żonę! Właśnie te gry chciałbym polecić wszystkim początkującym graczom. Są to:

CHUCK ROCK, TOKI, ELVIRA ARCADE GAMES, ORK, RUBICON, HARLEQUIN, SHADOW OF THE BEAST (cz.1 i 2), HUGO 2, NEWBULUS 2, NAVY SEALS, PEGASUS, MEGA TWINS, ZONE WARRIOR, WAR ZONE, UNDER PRESSURE, ELF, VOODOO NIGHTMARE.

Dla graczy nie gustujących w tego typu grach polecam coś do myślenia, czyli gry logiczno-zręcznościowe. Tu od razu polecam Wam całą serię LEMMINGS'ów. Są to wspaniałe gry, w których grupa stworków (nastawionych przyjaźnie do świata) wędruje sobie po świecie. Należy im w tym pomagać. Cała trudność ogranicza się do przesuwania krzyżyka na ekranie i podawania stworkom w odpowiednim czasie różnych przedmiotów. Dzięki owym przedmiotom Twój LEMMING może stać się koparzem w ścianie/ziemi, wspinaczem, spadochroniarzem czy też stoperem, który w razie niebezpieczeństwa powstrzyma lub zawróci całą gromadkę.

Inne gry logiczne-zręcznościowe to:

TETRIS - należy układać spadające cegiełki w taki sposób, aby pomiędzy nimi nie było żadnych wolnych przestrzeni.

TWINTRIS - to samo co tetris, tyle że dla dwóch graczy.

TILT - przesuwanie gotowych elementów, po których ma trafić do celu mała piłeczka.

BLOCKOUT - trójwymiarowe TETRIS (trudne!).

LOGICAL - spadające kulki w różnych kolorach, które trzeba grupować po cztery sztuki w specjalnych kołach połączonych korytarzami.

Jeśli natomiast i to Ci nie odpowiada, proponuję zagrać w symulatory, gry sportowe, strategiczne bądź też wszelkiego rodzaju „wybijanki" (bij-zabij). Wymienię najlepsze tytuły:

Symulatory samochodowe:

INDIANAPOLIS 500 - wyścigi formuły pierwszej.

LOTUS TURBO ESPRIT - kto pierwszy ten lepszy!

LOTUS TURBO ESPRIT CHALLENGE - dalszy ciąg LOTUSA.

TEST DRIVE 1 i 2 - sprawdzian z jazdy.

VROOM - po prostu szaleńcze wyścigi formuły 1!

INDY HEAT - jazda małą formułką (widok z boku, z góry, wspaniała grafika i muzyka).

Symulatory lotnicze:

BIRDS OF PREY, F-29 RETALIATOR, F-19, A-10 TANK KILLER, STORMOVIK, F-117A NIGHTHAWK, B-17 BOMBER, RED BARON.

Strategiczne:

POPULOUS 1 i 2, POWERMONGER 1 i 2, UTOPIA, MEGA LO MANIA, ASCHESS OF EMPIRE, BATTLE CHESS 1 i 2.

Wybijanki:

ORK, SWIR, WARZONE, VIDEO KID, APIDYA (strzelaniny).

TEAM YANKEE i SCHERMAN-M4 (wojsko naziemne).

SILENT SERVICE 2, 688 SUBMARINE ATTACK, ADVANCED DESTROYER SIMULATOR, HARPOON, WOLF PACK (statki, łodzie podwodne).

Karate:

FIRST SAMURAI, LAST NINJA 3, BUDOKAN, FINAL FIGHT, FULL CONTACT, SHADOW WARRIOR.

Sport:

KICK OFF 2, JIMMY WHITE SNOOKER, SPEEDBALL 2, PRO TENIS 2.

Tekstówki:

SECRET OFĽTHEĽMONKEY ISLAND, LARRY SUIT 5.

Przygodowe:

HARLEQUIN, ELF, SHADOW OFĽTHEĽBEAST 1 i 2, NEBULUS, DIABOLIK, GOTHA, ADAMS FAMILY, MAGIC POCKETS, IMMORTAL.

Osobiście bardzo lubię grać w różne gry, namawiam jednak wszystkich do wykorzystywania komputera do celów bardziej użytecznych. A jak zamierzacie tylko i wyłącznie grać, to kupcie sobie po prostu NINTENDO z modułami!

OLAF

LOGICAL

Historia zatoczyła koło. W czasach przedkomputerowych (mam tu na myśli komputery domowe) były popularne wszelkiego rodzaju układanki, takie jak Magiczny Kwadrat, czy jego odmiana Jajo Kolumba, nie mówiąc już o Kostce Rubika. Obecnie furorę robią komputerowe odmiany tego rodzaju gier, czego przykładem może być np. Tetris. Można się pokusić o stwierdzenie, że komputery „stworzyły" nową jakość w dziedzinie gier (układanek) logicznych, umożliwiając wprowadzenie ruchu jako elementu scenariusza.

Logical firmy Rainbow Arts jest jedną z gier tego typu. Reguły gry są bardzo proste, wpadające w kanale na samej górze kule należy umieszczać w specjalnie do tego celu przeznaczonych pojemnikach. Gdy w pojemniku spotkają się cztery kule jednego koloru pojemnik eksploduje (to chyba ma jakiś związek z przekroczeniem masy krytycznej).

Etap kończy się z chwilą, gdy przynajmniej raz „wysadzimy w powietrze" każdy pojemnik. Niby nie jest to nic trudnego i wszystkie 99 poziomów gry

można by przejść za jednym razem gdyby ... Otóż właśnie, pojemniki i łączące je kanały to nie wszystko, na co może natrafiać wędrująca kulka. Wśród różnych rozmaitości zdarzają się bramki przepuszczające tylko kule o określonym kolorze lub jednokierunkowe kanały. Szczególnie nieprzyjemne jest ich zestawienie. Kulka o niewłaściwym kolorze skierowana w drogę jednokierunkową zamkniętą bramką będzie „tłukła się" w labiryncie do końca etapu. Do przyjemnych niespodzianek należy mała „lakiernia" zmieniająca kolor przechodzącej pod nią kuli, co znacznie ułatwia zgromadzenie w jednym miejscu kul o jednakowym kolorze. W niektórych etapach można spotkać coś w rodzaju sygnalizacji świetlnej, którą należy „zgasić" w pierwszej kolejności, aby nie zablokować się pełnymi, a nie „wysadzonymi w powietrze" pojemnikami.

Szczytem utrudnień jest pojawiający się od czasu do czasu kolorowy kod, składający się z czterech różnokolorowych kul. Jest to swego rodzaju zapalnik. Póki nie ułożymy w jednym pojemniku podanego kodu, inne pojemniki prawidłowo wypełnione nie eksplodują. Jak by tego nie było dosyć, na wypełnienie zadania mamy ograniczoną ilość czasu. Limitowany jest nie tylko czas ukończenia etapu, ale i czas, przez który kula może poruszać się w górnym kanale.

Ukończenie wszystkich etapów gry nie oznacza, że Logical wędruje na półkę. Po ostatnim poziomie otrzymujemy hasło pozwalające na uruchomienie edytora poziomów. Tak więc zabawa toczy się dalej. Ja niestety nie mogę nic więcej na ten temat powiedzieć, bowiem na razie dotarłem do 68 poziomu.

Tych wszystkich, którzy chcieliby zagrać w Logical ostrzegam: gra wciąga. Należy zarezerwować sobie wolny dzień, lub lepiej dwa, na zabawę z Logical'em. Nie należy się łudzić, że można się od niego oderwać po godzinie.

PAWEŁ GALAS

FIRMA: **Rainbow Arts**
Dystrybutor:
IPS Computer Group, Warszawa
• rodzaj gry: logiczna
• komputer: Amiga, C-64, Atari ST, IBM
• wymagania: kolorowy monitor (wersja IBM)

PIT FIGHTER

Od pewnego czasu brooklyńskim półświatkiem zawładnął nielegalny, okrutny i bezwzględny sport, wzorowany na starożytnych walkach gladiatorów. Przyciśnięci przez biedę młodzi i silni mężczyźni za kilka dolarów decydowali się wziąć udział w walce, wiedząc, że mogą w niej stracić nawet życie. Takie „przedstawienia" zyskały wielką popularność także w wyższych sferach, głównie ze względu na możliwość zdobycia znacznych pieniędzy w zakładach.

Jesteś jednym z zawodników. Wraz z Tobą walczą jeszcze Kato, posiadacz czarnego pasa w karate, i Buzz, nadzwyczaj silny, były zapaśnik. Macie przeciwko sobie ośmiu przeciwników, z których każdy walczy w inny sposób.

Nie jesteś oczywiście bezbronny. Dozwolone są wszystkie chwyty, więc skorzystaj z różnych przedmiotów leżących na ziemi, dzięki którym możesz skuteczniej unieszkodliwić wroga.

Spotkania są tak zorganizowane, że co trzecią walkę toczysz z drugim graczem (w przypadku gdy grasz z kolegą) lub swoim sobowtórem (gdy grasz sam). Punkty premiowe przyznawane są pod koniec każdej rundy za nokauty (KO). Przedmioty, które znajdujesz na arenie to: Throwing stars (latające gwiazdy), Barrels (baryłki), Kegs (beczułki) i Sticks (dzidy). Każdy z wymienionych przedmiotów można użyć tylko raz. Wyjątkiem są dzidy, które pękają dopiero po kilku uderzeniach.

Warto pograć w tą grę, choćby po to, by sprawdzić swój refleks i budowę joysticka. Spróbuj zostać championem.

PIOTR LISZEWSKI

Firma: **DOMARK**
Dystrybutor:
IPS Computer Group, Warszawa
• komputer: C-64
• rodzaj gry: sportowo-zręcznościowa
• wymagania: -

GRAFIKA
MUZYKA
OGÓLNIE

GRAFIKA
MUZYKA
OGÓLNIE

Pit Fighter

EUROPEAN SPACE SIMULATOR

Jednym z największych ludzkich marzeń było oderwanie się od Ziemi. Najpierw powstał balon, nieco później samolot, a na końcu (jak dotychczas) wahadłowiec. Może w przyszłości będziemy odwiedzać sąsiadów z innej galaktyki na pokładach dalekosiężnych nadprzestrzennych gwiazdolotów; na razie jednak pozostaje to w sferze marzeń. Pozostańmy więc przy wahadłowcach.

Jeśli marzysz o wyprawie w Kosmos, możesz to uczynić dzięki EUROPEAN SPACE SIMULATOR. Oczywiście jest to tylko symulacja, ale jeśli ktoś ma wyobraźnię, to przypina się do fotela paskiem pożyczonym od starszego brata i startuje.

EUROPEAN SPACE SIMULATOR to gra biorąca swój początek w „kosmicznych" grach handlowych ze słynną „ELITE" na czele. Jedyna różnica polega tu na tym, że nikt nie zamierza Cię zaatakować, nie musisz do nikogo strzelać - jesteś po prostu pilotem kosmicznego promu (o nazwie TOTH), zarabiającego na wywożeniu w przestrzeń kosmiczną satelitów i fragmentów stacji badawczej. Najpierw musisz sobie skompletować załogę, następnie wybierasz rodzaj ładunku (pamiętając przy tym, że udźwig promu jest ograniczony).

Po wejściu na orbitę można ewentualnie zniszczyć laserem uszkodzonego satelitę lub wyskoczyć na małą wyprawę kosmicznym skuterem. Nie ma zbyt wielu opcji. Po zakończeniu misji czeka Cię lądowanie i... przygotowanie promu do następnego lotu.

Dźwięk - właściwie muzyki tu brak, ale nie ona jest istotna w programach tego typu. Grafika na poziomie Amigi, całość - w sumie - niezła.

BAD

Firma: **Tomahawk**
Dystrybutor:
JTT S.C., Wrocław.
• rodzaj gry: symulacyjna
• komputer: Amiga, IBM PC, Atari ST
• wymagania: -

GRAFIKA
MUZYKA
OGÓLNIE

P-47 Thunderbolt

P-47 THUNDERBOLT

Jeżeli pragniesz się na czymś wyżyć, to być może będzie Ci potrzebna „latająca" strzelanina. Jedną z gier tego typu jest właśnie P-47 THUNDERBOLT.

Poruszasz się w czterech kierunkach małym samolocikiem. Twoim zadaniem jest zestrzelenie wszystkiego, co się rusza i strzela. Oprócz wrogich samolotów i latających spodków wśród drzew i wzgórz kryją się nieprzyjacielskie czołgi i baterie przeciwlotnicze. Czasami, gdy trafisz wrogi samolot, na jego miejscu pojawi się kolorowa ikona, którą należy zebrać, dzięki niej otrzymasz nową broń, np. bomby, rakiety czy pociski rozpryskowe.

Pod koniec każdego poziomu pojawia się olbrzymia maszyna (samolot, czołg, okręt), którą należy unicestwić przez wielokrotne strzelanie w jej działo. Podczas wczytywania kolejnych poziomów na ekranie pojawia się rysunek z informacją, w jakim miejscu będziesz obecnie walczyć.

Grafika oraz muzyka nie należą do zachwycających, ale uwagę zwraca różnorodność krajobrazów przesuwających się za samolotem. Gra należy do rodziny zręcznościowych - wcale nie jest łatwa! W komplecie znajduje się dyskietka 5,25" z programem oraz instrukcja obsługi w formie plakatu reklamowego. Ogólnie rzecz biorąc gra, jak wiele innych, nie wyróżnia się niczym specjalnym, choć dla amatorów strzelania może być satysfakcjonująca.

PIOTR LISZEWSKI

Firma: **FIREBIRD**
Dystrybutor:
JTT S.C, Wrocław
• komputer: C-64
• rodzaj gry: zręcznościowa
• wymagania: -

GRAFIKA
MUZYKA
OGÓLNIE

Wspomnień artylerzysty część VIII
Przez J. E. Bartka Kachniarza w pięciu księgach prozą spisana

SPRAJTY SĄ Z GUMY

1. C-64 w porywach

W lipcu zaczęliśmy przerabiać nasz komputer na maszynę o potężnych możliwościach. To, do czego potrafimy zmusić naszego „komcia" coraz dalej wykracza poza to, czego dowiedzieliśmy się z instrukcji obsługi (dotyczy to oczywiście tych, którzy ją czytali - ja nie). Dlatego coraz częściej na zapytania znajomych „A jakie są możliwości tej maszynki?" odpowiadać będziemy „W zasadzie takie, a w porywach takie". Czyli: Ile można wyświetlić kolorów? W zasadzie 16, a w porywach za setkę. Ile C-64 ma sprajtów? W zasadzie 8, a w porywach do pół tysiąca. O ile można przesunąć ekran w dół? W zasadzie osiem pikseli, a w porywach - ile kto chce. Czy można wyświetlać sprajty na ramce? W zasadzie nie, a w porywach - czemu nie? Ile razy można poszerzyć sprajty w pionie? W zasadzie 2 razy, a w porywach - ooo, panie! One mogą zająć cały ekran!

2. Sprajty się rozciągają

Przy czym - zastrzegam - nie będziemy korzystać z niczego innego jak poczciwe poszerzanie podwójne w komórce $d017 (YXPAND). No, może tylko, jak zwykle, nieco ogłupimy nasz komputer. Cały trik polega na tym, że sprajta należy w każdej linijce zwęzić - czyli doprowadzić do normalnych rozmiarów - a potem (znienacka!) rozszerzyć. Brzmi głupio, nieprawdaż? Ale działa. Na jakiej zasadzie, dokładnie nie wiadomo, ale - jak za każdym razem - komuś udało się dorobić do tego filozofię. A brzmi ona tak: kiedy zwęzimy sprajta, to nie daje to efektu natychmiast, bo w jednej linii to i tak nie robi różnicy, czy sprajt jest rozszerzony, czy nie - i tak widać tylko jedną jego linijkę. Dopiero potem, od następnej linijki można się zastanawiać, jak ten sprajt ma właściwie wyglądać. A teraz (nie przeszedłszy do następnej linijki) znowu rozszerzamy sprajta. Też w tej samej linii nie daje to efektu, ale komputer konotuje sobie z cicha, że trzeba będzie tego sprajta rozszerzyć. W następnej linii - powtarza zawartość sprajta z linii poprzedniej. W końcu $d017 do czegoś zobowiązuje. I tu kółko się zamyka.

Właściwie jesteśmy w stanie rozszerzyć dowolną linię sprajta dowolnie dużo razy. Możemy potem z dumą informować osłupiałych atarowców, że oto właśnie rozszerzyliśmy sprajta dwieście razy, a te pionowe kreski, co je na ekranie widać, to po prostu pierwsza linia sprajta. Reszta się nie mieści. Jeśli nasz czcigodny interlokutor nagabywał będzie o resztę, to zawsze możemy go posłać po dwadzieścia monitorów (jak wróci ze sprzętem, to też nie ma strachu - po prostu poszerzymy sprajta nie 200 a 4000 razy i znów widać będzie tylko pierwszą jego linijkę - co za pech!).

Ale, by sztuczka nasza nie służyła li tylko do pokazywania pionowych kresek (bo taki efekt da nieskończone poszerzanie pierwszej linijki) musimy w jakiś sposób przejść do rozszerzania pozostałej części sprajta. Jak? Elementarne, drogi Watsonie: nie rozszerzając go w jednej linijce. Komputer dojdzie do wniosku, że sprajt jest zwężony, czyli w następnej linijce nie należy powtarzać tego, co było w poprzedniej, a po prostu przeskoczyć do następnej linijki sprajta. I po kłopocie.

3. Technika

Warto jeszcze tylko dodać, że korzystać będziemy z techniki FLD. Dzięki niej odsuniemy ekran, przez co znacznie łatwiej będzie nam docyklować procedurę. Jeśli chodzi o ścisłość, wcale nie będziemy musieli jej cyklować. Wystarczy nam sprawdzanie co linię przejście do linijki kolejnej. Powiedziałem już wszystko, co do powiedzenia było. Teraz czas na kwas, to jest na listing. Proponuję wpisać z uwagą PROGRAM 1. Ponad 1/3 całości zajmuje nam przygotowanie gruntu pod ostrzał. Ustawiamy przerwania, przygotowujemy sprajty i zapętlamy program.

Dygresja na temat zapętlania.

Zapewne wielu z was doznawało nie raz lekkiego szoku podczas wpisywania listingów artyleryjskich. W pewnym momencie prawie zawsze pojawiał się rozkaz w rodzaju:

1045 JMP $1045

Zaiste insteresujące, nieprawdaż? Na zdrowy rozum nie ma to sensu: komputer kotłuje ciągle w tej linijce i świata poza nią nie widzi. Diabeł, jak zwykle, siedzi w szczegółach. Nie wiem, czy zwróciliście uwagę, ale linia ta ZAWSZE poprzedzona jest rozkazem CLI. CLI ma to do siebie, że gasi maskę przerwań, dzięki czemu co 1/50 sekundy procesor dostaje sygnał, że wartałoby skoczyć do procedury, którą nazwaliśmy IRQ, a którą wskazuje wektor w komórkach $0314/$0315. W ogóle przerwania działają jak telefon w biurze: pracownik przekładający papiery i stukający na maszynie (jak ja teraz) odkłada swoją robotę, podnosi słuchawkę i słucha, co kto do niego mówi. Telefon to przerwania maskowalne - nasz urzędnik zawsze może wyciągnąć wtyczkę z gniazdka. Są też przerwania niemaskowalne - na przykład zaczyna płonąć firanka. Urzędnik rzuca wszystko w diabły i albo ucieka gdzie pieprz rośnie, albo zaczyna ją gasić resztkami herbaty. Daje to taki efekt jak naciśnięcie RESTORE albo RESET. Tych przerwań zamaskować się po prostu nie da. Nie da i już.

Zapętlanie służy więc nam do tego, by komputer cokolwiek robił. I żeby było dokładnie wiadomo, co robi. Poza tym jest jeszcze kwestia modułu, który siedzi u mnie w komputerze. Nazywa się FINAL III. I ma to do siebie, że cholernie chytrze przejmuje obsługę przerwań i nie ma za bardzo na niego siły. No nie ma. Jeśli zaś modułu takiego nie macie, możecie wstawić RTS i działać powinno bez oporów.

Następna część (oddzielona linią) to tzw. żywotna część całej imprezy. Tu leży tzw. sedno. Przelećmy szybko: przyjęcie przerwania (INC $d019), odczekanie do linii nr $40 ($40, bo w przygotowaniu ustawiliśmy duszka w linii $41). Potem rozkazy znane już z FLD. A teraz dochodzimy do części najistotniejszej. LDX #$00/STX $d017. Te rozkazy to po prostu informacja o kompletnym zwężeniu wszystkich dostępnych sprajtów. Ale zaraz potem ładujemy do rejestru X jakąś wartość z tabeli. Rzut oka do tabeli... Aha, to może być zero albo jedynka, po prostu. Jeżeli jest jedynka, to sprajt jest rozszerzany o jedną linię. Jeśli jest zero, to przechodzimy do linijki kolejnej, zaprzestając rozszerzania poprzedniej (dzięki tym zerom likwidujemy konieczność unikania nieskończonych pionowych pasów). Potem tylko powiększymy Y o 1 i, jeśli nie dojdzie do 255 (a co?! Jak szaleć to szaleć, panie!), to powtarzamy całą operację. Jeśli dojdzie, to jeszcze dla pewności znowu zwężamy sprajta do zwykłej szerokości i skaczemy do $ea81, co w krew zaczyna nam już wchodzić.

Uruchom teraz program. Dla przypomnienia tylko: TurboAssembler: strzałka w lewo, 3 i S. Poza tym programem: SYS 10000. I co widzimy? Łamaniec linii ukośnych i pionowych o długości ok. 1/3 ekranu. Przyjmij na wiarę, że cały ten łamaniec to jeden sprajt. Rozszerzany tu i ówdzie.

Jeszcze kwestia tabeli. Zera i jedynki wstawiajcie według własnego uznania. Obecny kształt tabeli jest spowodowany głównie tym, że programik ten powstawał po północy, a wtedy - jak wiemy - różne rzeczy do głowy przychodzą. Ten sam efekt (przychodzenia do głowy) widać też najczęściej w etykietach, których nazwy najczęściej sprowadzają się do d..a1, d..a2, d..a+++ i rano człowiek ma zagadkę, cóż one takiego oznaczają.

Oj, odszedłem od tematu. Jeżeli chciałbyś rozszerzyć sprajta, no, niech będzie trzykrotnie, to musisz wstawiać dwie jedynki, a po nich zero. I tak ciągle.

4. A może by więcej?

A może by nie ograniczać się do jednego sprajta, a rozszerzyć ich... aż trzy, tak na początek? Rzućmy okiem na LISTING 2. Krótkie porównanie z pierwszym... no cóż, największa różnica wielkości widoczna jest w części pierwszej, tj. przygotowaniu. Po prostu mamy trzy razy więcej sprajtów do ustawienia. Więcej też zajmuje przygotownie matrycy duszka, bardziej nieco fantazyjnej. Część II (czyli tzw. sedno) pozostaje BEZ ŻADNYCH ZMIAN po prostu. Rewolucja za to odbyła się w tabeli. Ach, zera i jedynki odeszły w niebyt... wyroiło się szóstek, trójek i tym podobnych. O tym na razie jednak sza. Uruchom program, jak leci. I co widzisz? Trzy desenie mające na sobie wyraźne piętno potraktowania moim programikiem. Rozciągnięte. Ale... Przyjrzyjmy się im bliżej. Ten najszerszy jest w tym miejscu, ten tam, tamten znów gdzie indziej. Czyli... (chwila namysłu) ...rozszerzane sprajty są od siebie całkowicie niezależne!

Teraz zajmijmy się tabelą. 6 to binarnie 00000110. 7 to 00000111, 5 to 00000101 i tak dalej. Już wiemy? Każdy sprajt ma swoją własną jedynkę, i jej tylko się trzyma. Dlatego właśnie rozszerzać je możemy oddzielnie.

5. Uwagi końcowe

Teraz w zasadzie wiesz już wszystko. Wiesz na czym efekt polega i w ja-

ki sposób stosować go do paru sprajtów. I tu zostaje pole dla Twojej inwencji. Zrób coś, by tych sprajtów było osiem, potem zrób tak, żeby tabele miały jakiś sens, a nie były aż do tego stopnia przypadkowe a następnie jeszcze zrób tak, żeby tablice mogły się zmieniać w czasie. Żeby sprajty się wesoło rooooozciągały i kurczyły. To są już sprawy proste, a robiące dość małą różnicę w listingach. Oczywiście, mógłbym je wszystkie tu umieścić, ale nie podejrzewam, żeby miało to aż tak wiele sensu.

Oj, zaczynam kwasić. Pora wiać.

BARTEK KACHNIARZ

LISTING 1

```
        *= 10000

        sei
        lda #$01
        sta $d01a
        lda #$7f
        sta $dc0d
        lda #<irq
        sta $0314
        lda #>irq
        sta $0315
        lda #$f9
        sta $d012
        lda #$1b
        sta $d011
        lda #$41
        sta $d001
        lda #$80
        sta $d000
        lda #$01
        sta $d015
        ldx #$00
sprajt  sta $0340,x
        sta $0358,x
        sta $0370,x
        inx
        inx
        inx
        asl a
        bne sprajt
        lda #$0d
        sta $07f8
        cli
klaps   jmp klaps
;
irq     inc $d019
        lda #$40
czekaj  cmp $d012
        bne czekaj
        ldy #$00
linia   lda $d012
wdol    cmp $d012
        beq wdol
        and #$0f
        ora #$18
        sta $d011
        ldx #$00
        stx $d017
        ldx tabela,y
        stx $d017
        iny
        cpy #$ff
        bne linia
        lda #$00
        sta $d017
        jmp $ea81
        .byte $07
;
tabela.byte 0,1,0,1,1,0,1,1,1
.byte 0,1,1,1,1,0,1,1,1
.byte 1,1,0,0,1,1,1,0,0
.byte 1,1,1,1,1,1,1,1,1
.byte 1,1,1,1,1,1,1,1,1
.byte 1,1,1,1,1,1,1,1,1
.byte 1,1,1,1,1,1,1,1,1
.byte 1,1,1,1,1,1,1,1,1
.byte 0,1,1,0,1,1,1,1,1
.byte 0,1,1,1,0,1,1,1,1
```

LISTING 2

```
        *= $2710
        sei
        lda #$01
        sta $d01a
        lda #$7f
        sta $dc0d
        lda #<irq
        sta $0314
        lda #>irq
        sta $0315
        lda #$f9
        sta $d012
        lda #$1b
        sta $d011
        lda #$41
        sta $d001
        sta $d003
        sta $d005
        lda #$c0
        sta $d002
        lda #$80
        sta $d000
        lda #$40
        sta $d004
        lda #$ff
        sta $d015
        ldy #$00
        ldx #$00
        pha
staw    tya
        sta $0340,x
        pla
        sta $0341,x
        eor #$ff
        pha
        iny
        inx
        inx
        inx
        cpx #$3f
        bne staw
        pla
        lda #$0d
        sta $07f8
        sta $07f9
        sta $07fa
        cli
klaps   jmp klaps
;
irq     inc $d019
        lda #$40
ty      cmp $d012
        bne ty
        ldy #$00
eee     lda $d012
eef     cmp $d012
        beq eef
        and #$0f
        ora #$18
        sta $d011
        ldx #$00
        stx $d017
        ldx tabb,y
        stx $d017
        iny
        cpy #$ff
        bne eee
        lda #$00
        sta $d017
        jmp $ea81

tabb .byte 6,7,4,3,7,6,5,3,7
.byte 2,1,3,7,7,6,7,7,7
.byte 1,1,2,2,1,1,3,3,3
.byte 3,3,3,3,3,3,2,2,3
.byte 1,1,3,3,3,3,3,1,3
.byte 3,3,3,1,3,1,3,3,3
.byte 1,3,3,3,3,3,3,3,3
.byte 1,1,1,1,1,1,1,1,1
.byte 0,1,1,0,1,1,1,1,1
.byte 0,1,1,1,0,1,1,1,1
0
```

Tytuł / Liczba kolejnych zeszytów	3	6	12	po ile egz.
	X	60000	120000	
	30000	60000	X	
	27000	54000	X	

Niniejszy kupon jest ważny do:

30-12. 1992

tu zanotuj, co zamówiłeś

Warunki prenumeraty:

- Prenumerata zawarta przed upływem ważności kuponu gwarantuje niezmienność cen
- Przesyłka pocztowa nie wymaga dodatkowych opłat
- Jeżeli w ciągu 2 tyg. od pojawienia się numeru w kioskach przesyłka nie nadeszła, prosimy o kontakt
- Za błędy wynikające z niestarannego wypełnienia formularza redakcja nie ponosi odpowiedzialności
- Prosimy o staranne i wyraźne zakreślenie odpowiednich ilości egzemplarzy

NINIEJSZYM ZAMAWIAM:

m-c	1	2	3	4	5	6	7	8	9	10	11	12
„C&A”												
Dyskietka 5.25” dla C-64/128/16/116/+4												
Dyskietka 3.5” dla Amigi												

Kaseta dla C-64 ☐

IMIĘ ....

NAZWISKO ....

ADRES ....

.... (Kod): ....

UWAGI:

KUPON WAŻNY DO 31. 01. 1993

Do kuponu dołączam kopię dowodu wpłaty.

NAJPIERW PRZECZYTAJ, POTEM ZAMAWIAJ!

1. Kupon ułatwia i przyspiesza realizowanie Twoich zamówień.
2. Kupon bez dowodu wpłaty (lub kopii) jest tylko nic nie wartym papierkiem. Zawsze, zanim wyślesz do nas zamówienie, upewnij się, czy włożyłeś do koperty **kupon i kopię dowodu wpłaty.**
3. Zaznacz krzyżykiem w odpowiednim miejscu to, co zamówiłeś. Jeśli Twoje zamówienie jest nietypowe, masz miejsce w rubryce „UWAGI”, aby wszystko dokładnie wyjaśnić.
4. Podawaj zawsze **PEŁNY** adres, nie zapominaj o kodzie pocztowym, numerze ulicy itp.
5. Wypełnij kupon **WYRAŹNIE!**

Ceny:

Numery archiwalne „C&A”:	12.000 zł/szt.
Dyskietki 5,25” na C-64/128/16/116/+4	30.000 zł/szt.
Dyskietki 3,5” na Amigę:	30.000 zł/szt.
Kaseta na C-64:	100.000 zł/szt.

(na kasecie nagrane są wszystkie programy dla C-64 publikowane w „C&A” nr 2-6/92)

Pieniądze należy wpłacać dla:
Spółdzielnia „Bajtek”
Warszawa, ul. Wspólna 61
na konto:
Bank „Agrobank S.A.”
470005-1834-131
ul. Grochowska 262
04-398 Warszawa

Jak zapewne zauważyliście, podnieśliśmy cenę dyskietek. Cóż inflacja dopadła i nas. Żywimy jednak cichą nadzieję, że nowa cena nie odstraszy Was zbytnio i że w dalszym ciągu będziemy dobrymi znajomymi.

Na listy do redakcji odpowiadać będzie stary spec, Bartek Kachniarz (który ma jednak to do siebie, że nie jest Genialny w stopniu wystarczającym, by udzielać jedynie słusznych odpowiedzi na wszystkie zadane pytania).

● Po komendzie PRINT FRE (0) wychodzi mi wynik ujemny. Wiem co wtedy trzeba zrobić, ale czytałem, że jest to wynik błędu powstałego po przeniesieniu bez zmian fragmentów systemu operacyjnego z innych komputerów Commodore (nikomu nie pożyczałem komputera, a ten błąd ukazuje się nawet tuż po włączeniu komputera, bez pisania wcześniej jakiegokolwiek programu). Jak to się mogło stać? Po włączeniu komputera sprawdziłem zawartość pamięci BASIC'a i wyszedł wynik 38909 (nie pisałem wcześniej żadnego programu!). Gdzie pozostałe bajty?!!

Paweł

Na początku muszę Cię uspokoić. Nie denerwuj się - nikt nie ukradł Ci bajtów ani nie podmienił cichcem systemu operacyjnego w komputerze. Autor artykułu miał na myśli leniwych programistów firmy Commodore, którym nie chciało się pisać od podstaw systemu operacyjnego dla C-64 i użyli wersji starszej, stosowanej w komputerach serii VIC i PET. Jakie są tego skutki - widać. „Dzięki" temu, aby sprawdzić, ile pamięci jeszcze nam pozostało, najlepiej będzie napisać:

PRINT PEEK (51)+PEEK (52)*256-PEEK (49)-PEEK (50)*25 co da nam efekt w postaci właśnie 38909, gdy pamięć komputera jest całkowicie czysta. Co więc stało się tym razem? To proste. Komputer C-64 ma 64 KB RAM. Jednak część pamięci jest dla BASIC'a niedostępna dlatego, że w normalnych warunkach jest maskowana przez identycznie adresowaną pamięć ROM. Pamięć tę można wyłączyć, ale spowoduje to brak zrozumienia języka BASIC oraz kompletną „głuchotę i ślepotę" komputera. Swoją drogą, to bardzo chytry chwyt ze strony projektantów - adresowane jest tylko 64 KB pamięci (i więcej być nie może!) podczas, gdy w rzeczywistości C-64 ma jej ni mniej, ni więcej tylko 88 KB. Adresowanie pamięci powyżej 64 KB musi być wspomagane czynnością nazywaną bankowaniem, czyli przypisywaniem różnym obszarom pamięci tego samego adresu i rozróżnianiem ich za pomocą specjalnego rejestru kontrolnego.

● Czy można tłumaczyć język asemblera na BASIC i odwrotnie, jeżeli tak, to podajcie tytuły programów, które to umożliwiają to przeprowadzić.

W jaki sposób można napisać program w języku asemblera, tzn. za pomocą jakiego programu (tytuł) można go nagrać na kasetę wraz z BASIC'em, w celu uruchomienia go procedurą (SYS XXXX). Chciałbym programować w jęz. maszynowym lecz nie wiem, gdzie są rozmieszczone poszczególne obszary pamięci takie jak np. pamięć przeznaczona na BASIC czy też na program w języku maszynowym.

Marcin

Przekładanie asemblera na BASIC jest na C-64, o ile mi wiadomo, niemożliwe. BASIC jest zbyt mało elastyczny, by wiernie oddać strukturę programów napisanych w asemblerze. Wykonanie każdej instrukcji BASIC wymaga kilkunastu (lub kilkudziesięciu) rozkazów asemblera, więc próby takie powodowałyby potworny spadek prędkości programów.

Zaś odwrotnie też nie jest łatwo, ale częściowo można praktykować. Programy tłumaczące (kompilujące) BASIC na asembler nazywamy kompilatorami. Powodują one pewne zwiększenie prędkości programów, ale i tak są znacznie wolniejsze od programów w „czystym" asemblerze. Przykładowe kompilatory BASIC na C-64 to: BASIC 64, BLITZ!, AUSTROCOMPILER i PETSPEED.

Nagrać program w asemblerze na dysku tak, by miał uruchamiającą linię w BASIC można np. przy pomocy programów zwanych LINKERAMI. Linkery najczęściej połączone są z programi pakującymi - KOMPRESORAMI. Potraktowanie programu w asemblerze kompresorem daje nam podwójną korzyść - raz, że można go będzie uruchomić zwykłą instrukcją RUN, dwa - znacznie zmniejszy się jego długość.

Ogólnie rzecz biorąc, pamięć nie jest podzielona: tu BASIC, tam asembler. Przyjęło się jednak, że programy w BASIC'u umieszcza się od adresu $0801 (2049), zaś krótkie procedury w asemblerze - od $c000 (49152). Oczywiście, obie te wartości nie są bezwarunkowo obowiązującymi. Każdy może umieszczać programy, gdzie mu wygodniej.

● 1. Czy warto dokupić stację dysków do C-64? O jakie nowe możliwości wzbogaca ona ten komputer? Czy stacja 1541-II jest odpowiednia?

2. Czy napęd dyskowy daje C-64 nowe możliwości graficzne? Czy są one widoczne w porównaniu z grafiką gier „magnetofonowych" (chodzi mi o porównanie gier z dysku i gier na taśmie)?

3. Czy istnieje digitizer lub podobne urządzenie do C-64?

4. Czy C-64 ma możliwość współpracy z magnetowidem lub kamerą wideo? Jeśli tak, to jak można zastosować magnetowid i jakie są wyniki tego zastosowania? Jakie programy są potrzebne?

Tomasz

Ad. 1. To zależy. Jeżeli komputer służy jako podstawka pod kurz, to nie warto. Jeżeli zaś jest w ciągłej eksploatacji - zdecydowanie tak! Stacja dysków pozwala na znacznie wygodniejszy dostęp do danych niż magnetofon. Przewagi stacji nad magnetofonem są dość liczne, a najważniejszą jest fakt, że jeśli jakiś program jest na drugim końcu kasety, to trzeba ją przewijać, dokładnie ustawić licznik oraz zajmować się setką podobnych dyrdymałów, które komputer zrobi za nas znacznie szybciej, gdy ma do dyspozycji stację dysków.

Ad. 2. Pytanie dość kłopotliwe. W pierwszym odruchu chce się wykrzyknąć: NIE! Mimo to jednak dalsza część pytania lepiej precyzuje problem. Należy więc powiedzieć, że gry całodyskowe są z reguły znacznie lepsze od wersji kasetowych. Nie wynika to ze zwiększenia możliwości komputera lecz z traktowania stacji jako specyficznego rozszerzenia pamięci. W wersji kasetowej ciężko jest np. pomieścić kilkanaście obrazków, do tego dużo animacji, muzyczek itd. Na dysku zaś nie stanowi to problemu.

Ad. 3. Jako digitizer rozumieć można urządzenie służące do przetwarzania obrazów lub dźwięków w ich cyfrowe odpowiedniki w pamięci komputera. Urządzenie do „scyfrzania" obrazów nazywa się skanerem. O skanerach do C-64 wiem tylko tyle, że ktoś je kiedyś produkował. Prawdopodobnie można je nabyć na Zachodzie.

Nieco popularniejsze są „scyfrzacze" dźwięku, czyli samplery. Samplery są w Polsce w niewielkich ilościach produkowane i czasem można je nabyć na giełdach komputerowych za ok. 200-300 kzł (kilozłotych).

Ad. 4. Ewentualna współpraca C-64 i magnetowidu jest, rzekłbym, jednostronna. Można bowiem podłączyć komputer do wideo za pomocą kabla antenowego (RF), dostroić magnetowid do C-64 jak do zwykłej stacji telewizyjnej, a potem nagrywać na taśmę obrazy przesyłane z komputera. W drugą stronę jest już znacznie gorzej. Potrzebna jest bowiem przystawka umożliwiająca „scyfrzanie" obrazu z magnetowidu. Urządzenia takie widziałem w wersjach dla Amigi lub PC, ale - niestety - nigdy nie słyszałem o ich odpowiednikach dla C-64.

Czy moglibyście mi podać po kolei co mam zrobić, żeby działały programy z siódmego numeru C&A na stronie 32. Gdyż jak wpiszę pierwszą linię I nacisnę RETURN, to na ekranie pojawia się napis SYNTAX ERROR.

Wojciech

Zaiste, moglibyśmy. Jeżeli chcesz wpisać którykolwiek z tych programów (za wyjątkiem PROGRAMU 2), musisz wczytać do pamięci i uruchomić program MONITOR. Program ten umożliwia wpisywanie tak właśnie podanych listingów. Dotyczy to - w ogólności - wszystkich listingów, których linie nie zaczynają się od numeru, jak Bóg przykazał, ale od kropki i dwukropka (np. .:19a0 0b 15 12 17 01 4d 41 43), kropki i litery A, lub samej litery A (np. A1000 LDA #$00). Dla większej wygody, monitory są zainstalowane w modułach typu FINAL II i III, ACTION REPLAY, EXPERT etc.



Takie właśnie rysunki składają się na SlideShow'y

dem-trader. Aby zrozumieć, co robi sysop, musicie przede wszystkim zrozumieć, co to są BBS'y. Otóż już wcześniej wspomniałem, że modem-trader to człowiek, który upload'uje i download'uje z BBS'u. A cóż to takiego?

BBS'y są to jakby takie „banki programów". Dzwoniąc do BBS'u mamy do dyspozycji cały SZTYWNY (twardy dysk) najnowszych programów (oczywiście dostępu do tych programów nie ma pierwszy lepszy śmiertelnik). Ilość programów w BBS'ach jest bardzo różna (tak jak pojemność twardych dysków) od 20 MB (bardzo rzadko - niezwykle nędzne BBS'y) do 2,4 GIGABAJTÓW!! (to raczej tyczy się tych najlepszych). Co do prędkości transmisji danych, to jest (na zachodzie) wśród grup amigowskich pewien ogólnie przyjęty standard - modemy HST (prędkość 14400 bodów), chociaż obecnie zaczyna wchodzić nowa prędkość (16800), a na przyszłość zapowiadana jest 28800 bodów. Dla przypomnienia 1 bod - 1 bit/s, 8 bodów - 1 bajt na sekundę, czyli 8192 body - 1 kilobajt na sekundę. Oczywiście prędkość ta dotyczy połączeń optymalnych (bez zakłóceń). Praktycznie jest ona o wiele niższa (szczególnie, jeżeli mamy do czynienia z „rewelacyjną" polską linią telefoniczną). No. ale wracając do tematu, czyli do sysop'ów. Czy domyślacie się już kto to może być? Tak, to jest człowiek będący posiadaczem owego BBS'a. To ten, który go obsługuje i dba, aby się doń nie zalogował (wpisał na listę użytkowników) nieodpowiedni człowiek. Sysop bowiem decyduje o tym, kogo przyjmie jako użytkownika i komu da jakie ratio (o tym za chwilę), kto na pewno będzie mógł się zalogować w dobrym BBS'ie, a kto na pewno nie. Jeżeli masz modem HST, to na pewno nie zostaniesz wyrzucony od razu, a czeka cię najwyżej rozmowa z sysop'em, od której zależy, jakie będziesz miał ratio. Ratio (z ang. przełożenie) jest to nic innego jak stosunek programów download'owanych do upload'owanych. Bo przecież naturalną sprawą jest, że dany użytkownik BBS'u nie może jedynie download'ować programów chyba, że sysop da mu „disabled", czyli możliwość nieograniczonych download'ów. Coś takiego dostać może jednak jedynie zasłużony obywatel (członek grupy, bardzo znana osobistość itd.), przeciętny użytkownik musi też trochę „zaapłołdować", aby móc ściągnąć dajmy na to najnowszą gierkę - LOTUS III.

Jeżeli posiadamy modem np. 2400 (modemy HST mają możliwe tylko dwie prędkości - 14400 i 2400, z tego też powodu z większością BBS'ów nie można się połączyć z prędkością np. 9600 bodów), to w zasadzie nie mamy po co dzwonić do dobrych BBS'ów (chyba, że lubimy oglądać komunikaty typu „sorry, you're lamer". Jeżeli natomiast mamy trochę szczęścia (i nieźle gadane, a w zasadzie pisane). to nawet przy prędkości 2400 możemy zagadać z sysop'em, aby się nad nami zlitował i pozwolił nam się zalogować. Jest to jeszcze uzależnione od tego, ilu liniowy jest dany BBS. Tak, nie mylicie się, dobre BBS'y działają na kilka linii (numerów) jednocześnie. Do rekordzistów należą tu amerykańskie BBS'y np. siedmioliniowy UNLAWFUL ENTRY grupy SKID ROW. Zresztą jest już ogólnie przyjęte, że BBS'y amerykańskie są najlepsze (najszybsze). Pod tym względem nie wiedzieć dlaczego cała Europa jest „lame".

LAMER. Bardzo wielką trudność sprawia mi wytłumaczenie tego najpopularniejszego słowa slangu. W zasadzie jest to temat na oddzielny artykuł. Why? Otóż słowo to jest największą obelgą dla amigowca związanego ze sceną i dlatego też jest ono bardzo nadużywane, a jego definicja staje się coraz bardziej płynna. Spróbuję jednak w kilku zdaniach napisać, kogo nazywa się lamerem. LAME po angielsku znaczy ciepły, a po amerykańsku kiepski, przeciwieństwem tego jest słowo COOL - zimny, a inaczej wspaniały, niemalże boski. Słowa te przyjęły się wśród amigantów całego świata i w świecie sceny nabrały konkretnego znaczenia. Lamerem nazywa się więc:

1) Wszystkich ludzi nie związanych ze sceną (czyli was również, drodzy czytelnicy, jeżeli nie zaczęliście jeszcze tworzyć na amidze).

2) Ludzi, którzy starają się być „cool elite", ale niestety ich produkcje są kiepskie (są początkujący, albo po prostu nie mają talentu).

3) Ludzi, którzy używają demomaker'ów (gotowych programów do robienia demosów), np. RSI DEMOMAKER 2.0.

4) Tych, którzy osiągają sukcesy w sposób nieuczciwy, tzn. przerabiają cudze grafiki i podpisują się pod nimi, a w demkach zrobionych pod demomaker'em podpisują się, że są wspaniali i w ogóle, robią demosy w oparciu o powyciągane procedurki i podpisują się pod nimi. Celowo do każdego z tych przykładów dodałem „podpisują się pod nimi", gdyż moim (prywatnym) zdaniem to właśnie jest lamerstwem.

A oto moja definicja lamerstwa: LAMER to osobnik, który uważa się za członka elity, w rzeczywistości nie mając z nią nic wspólnego. Bowiem sądzę, że RSI DEMOMAKER itp. jest dla ludzi, muzyka nie musi być od razu wspaniała, a kod na poziomie PHENOMENY. Ważne jest, by uczciwie traktować swoje osiągnięcia.

* * *

No tak, teraz po kruuuuciiiuuutkiiiim przerywniku możecie przeczytać, jak wygląda scena w chwili obecnej. Nasza opowieść skończyła się momencie powstania pierwszych grup robiących demka. Co było potem? Otóż niewiele później niektórym grupom posiadającym dobrych koderów, grafików i muzyków skończyły się dopływy nowych oryginałów (np. THE SILENTS) i grupy takie postanowiły zająć się jedynie robieniem demosów. W innych z kolei grupach były wewnętrzne zgrzyty, gdyż np. niektórzy graficy czy muzycy nie chcieli ryzykować swojej przyszłej kariery (a praca dla grup crackerskich mogła takową przekreślić). W ten sposób na przestrzeni mniej więcej roku doszło do rozdzielenia dwóch głównych nurtów sceny: CRACK i DEMO. Ale co to były wtedy za grupy! Najlepsze crack'i pochodziły od QUARTEX (!!!), ACKERLIGHT czy też FAIRLIGHT. A kto robił najlepsze demka? Sorry, ale wtedy jeszcze nawet nie miałem Amigi, a sławne były już takie grupy jak WILD COPPER CREW. PHENOMENA (chyba najstarsza grupa amigowska), FIRLIGHT (tak, ten sam co crackował gierki), a także SCOOPEX i BYTERAPPERS. Jak widać, niektóre z tych grup przetrwały do dzisiaj (choć w zmienionym niemalże całkowicie składzie).

Ponieważ demek nie powinno się robić tak bez sensu (a fajnie byłoby też coś na nich zarobić), więc zaczęto organizować COPY-PARTY. W zasadzie początkowo nie chodziło o nagrody, a jedynie o możliwość spotkania się w jak największym gronie maniaków. Pomysł chwycił i stał się dla wielu najwspanialszym aspektem sceny. Organizacji COPY-PARTY podejmowały się coraz to większe grupy i prześcigały się wzajemnie w organizacji (wielkość, jakość, suma nagród). Na party zjeżdżała się kupa ludu z komputerami w plecakach. Każda grupa pokazywała na co ją stać. Elitarne grupy mogły znaleźć „nowe młode talenty" i powiększyć swoje składy. Inne grupy mogły się rozsypać, jeszcze inne wejść we wzajemne kooperacje itd. Jednym słowem na party mogło zdarzyć się wszystko (no, prawie).

Dla maniaka Amigi party to jednak mało. Chce on czytać newsy ze sceny dzień i noc. Grupy chcą porównywać swoje demka do innych. Najlepsi chcą, aby pisano o nich jako o najlepszych itd. W ten oto sposób (może jeszcze wcześniej niż copy-party, choć któż pamięta, co było wcześniej) zaczęły powstawać magazyny dyskowe redagowane przez ludzi ze sceny. W magazynach tych porusza się (bo w tej chwili takowe również istnieją) wszystkie tematy mogące zainteresować scenę. Można tam wyczytać, kto przeszedł z jakiej grupy do jakiej, jaka grupa zrobiła jakie demo, wywiad z MR.X, ogłoszenia swapperów (w celu wymiany new stuff'u + friendship'u - przyjacielski kontakt), opisy sprzętu, a także głupawe dowcipy (żeby kto nie pomyślał, że amigerzy nie mają poczucia humoru), testy pizzy z włoskiej pizzerii czy BIG MAC'a z MAC DONALDA, a także rozważania na temat odwiecznych problemów sceny: kto to jest lamer, dlaczego jest tak mało dziewczyn na scenie i dlaczego Amiga jest lepsza od IBM PC (bo chyba każdy wie, że jest), oraz oczywiście najważniejsze, czyli charts'y. W tym przypadku w miarę szybko pałeczkę przejęła grupa CRUSADERS ze swoim EURO-CHARTS. Co to jest? Taki jakby TOP 20 (podobnie jak w MTV) tylko, że zamiast teledysków są najlepsze grupy, dema, muzycy, graficy, koderzy, magazyny dyskowe oraz oczywiście gierki. Pierwszym magazynem dyskowym na Amidze był „Cracker Journal", jednak w momencie pojawienia się takich magów jak „ZINE" stracił on znacznie na popularności (konkurencja była lepsza). Warto tu dodać, że Polska nie pozostaje w tej dziedzinie gorsza. Pierwszym polskim magiem był KEBAB (wszystkie magazyny dyskowe za wyjątkiem paru potentatów typu „ZINE", są darmowe), jednak szybko uległ komercjalizacji. Obecnie wydawany jest on na papierze i zupełnie nie przypomina swojego poprzednika (może nazwą). W tej chwili wydawane są nast. polskie magazyny dyskowe (kolejność alfabetyczna):

- **FAT AGNUS** wydawany przez grupę INVESTATION z Wrocławia,
- **NEXT LIFE** wydawany przez grupę FUTURE REVOLUTION z Częstochowy,
- **WHITE PAPER** wydawany przez grupę LUZERS z Gdańska i
- ZIG ZAG wydawany przez grupę ACTION DIRECT z Warszawy (tak się głupio składa, że jestem w tej grupie, a tak w ogóle, to jestem redaktorem naczelnym tego maga, ale to nie ważne). Ważne jest to, że wreszcie Wy - wspaniali Czytelnicy - będziecie mogli poczytać artykuły, jakie do tej pory można było przeczytać w MAGazynach dyskowych. Rozstajemy się bez „płyty" i zakończenia (bo i po co?), ale za to z niezbędną porcją wiedzy, dzięki której od tej pory będziecie mogli „załapać" artykuły o scenie i, miejmy nadzieję, nieco inaczej spojrzycie na demka.

O ile czas pozwoli, w następnym numerze zajmiemy się dokładniej samymi demkami.

NINJA/ACTION DIRECT